1991

Approximate confidence limit procedures for complex systems.

Yee, Kah-Chee

Monterey, California. Naval Postgraduate School

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

APPROXIMATE
CONFIDENCE LIMIT PROCEDURES
FOR COMPLEX SYSTEMS

by

YEE, Kah-Chee

September, 1991

Thesis Advisor: W. M. WOODS

Approved for public release; distribution is unlimited.

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED		1b. RESTRICTIVE MARKINGS	
2a. SECURITY CLASSIFICATION AUTHORITY		3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution is unlimited.	
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE			
4. PERFORMING ORGANIZATION REPORT NUMBER(S)		5. MONITORING ORGANIZATION REPORT NUMBER(S)	
6a. NAME OF PERFORMING ORGANIZATION Naval Postgraduate School	6b. OFFICE SYMBOL *(If applicable)* OR	7a. NAME OF MONITORING ORGANIZATION Naval Postgraduate School	
6c. ADDRESS *(City, State, and ZIP Code)* Monterey, CA 93943-5000		7b. ADDRESS *(City, State, and ZIP Code)* Monterey, CA 93943-5000	
8a. NAME OF FUNDING/SPONSORING ORGANIZATION	8b. OFFICE SYMBOL *(If applicable)*	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER	
8c. ADDRESS *(City, State, and ZIP Code)*		10. SOURCE OF FUNDING NUMBERS Program Element No / Project No / Task No / Work Unit Accession Number	

11. TITLE *(Include Security Classification)*
APPROXIMATE CONFIDENCE LIMIT PROCEDURES FOR COMPLEX SYSTEMS

12. PERSONAL AUTHOR(S) YEE, Kah-Chee

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT *(year, month, day)*	15. PAGE COUNT
Master's Thesis	From To	1991, September	129

16. SUPPLEMENTARY NOTATION
The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

17. COSATI CODES			18. SUBJECT TERMS *(continue on reverse if necessary and identify by block number)*
FIELD	GROUP	SUBGROUP	CONFIDENCE LIMIT PROCEDURES, COMPLEX SYSTEMS, LIFE TESTING, SERIES SYSTEMS, SYSTEM RELIABILITY RELIABILITY

19. ABSTRACT *(continue on reverse if necessary and identify by block number)*

Lower confidence limit estimation procedures for the reliability of several systems are developed and their accuracies evaluated using computer simulation. The procedures use test data on components of the system which can have failure times with either exponential or Weibull distributions or both. Testing scenarios for the components can be truncated by number of failures or by planned test times.
Although the evaluation effort was focussed on series systems in this thesis, the procedures readily apply to other systems as described in the thesis. The evaluations demonstrate the procedures to be quite accurate when sufficient component testing is performed.
Two FORTRAN computer programs were written to perform the evaluation. They are annotated in Users' Guides and can be used to determine the accuracy of these approximate lower confidence limit procedures for a given specific system and associated set of input parameters.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS REPORT ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED	
22a. NAME OF RESPONSIBLE INDIVIDUAL W. M. Woods	22b. TELEPHONE *(Include Area code)* (408)646-2743	22c. OFFICE SYMBOL OR/Wo

DD FORM 1473, 84 MAR 83 APR edition may be used until exhausted
All other editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

Approximate Confidence Limit Procedures
For Complex Systems

by

YEE, Kah-Chee
Operations Analyst, Ministry of Defence, SINGAPORE
B.Eng(Mech), National University of Singapore, 1987

Submitted in partial fulfillment
of the requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the
NAVAL POSTGRADUATE SCHOOL
September 1991

ABSTRACT

Lower confidence limit estimation procedures for the reliability of several systems are developed and their accuracies evaluated using computer simulation. The procedures use test data on components of the system which can have failure times with either *exponential* or *Weibull* distributions or both. Testing scenarios for the components can be truncated by number of failures or by planned test times.

Although the evaluation effort was focussed on *series* systems in this thesis, the procedures readily apply to other systems as described in the thesis. The evaluations demonstrate the procedures to be quite accurate when sufficient component testing is performed.

Two FORTRAN computer programs were written to perform the evaluation. They are annotated in Users' Guides and can be used to determine the accuracy of these approximate lower confidence limit procedures for a given specific system and associated set of input parameters.

THESIS DISCLAIMER

The reader is cautioned that the computer programs developed in this research may not have been exercised for all cases of interest. While every effort has been made, within the time available, to ensure that the programs are free of computational and logic errors, they cannot be considered validated. Any application of these programs without additional verification is at the risk of the user.

TABLE OF CONTENTS

ACKNOWLEDGEMENT

I am grateful to my thesis advisor, Professor Woods, for his trust and patience. His constant words of encouragement and timely advice is deeply appreciated. I wish to thank Professor Read for his help in editing this report. The staff of the NPGS computer center and the OR Department's microcomputer lab has been most courteous and helpful.

Finally, I must thank my wife, Jacqueline, for her jovial spirit and consistent support during our stay here in Monterey.

I. INTRODUCTION

This thesis develops approximate lower confidence interval procedures for the reliability of complex systems using test data on components of the system. The accuracies of these procedures are also assessed using computer simulation. The procedures can be used for any complex system whose reliability does not decrease when the reliability of any one of the components is increased.

The failure times of the continuously operating components are assumed to have either an *exponential* or *Weibull* distribution. Parameters in both distributions are assumed to be unknown. The *Weibull* distribution is used to model the lifetime probability distribution of electronic components with non-constant failure rate functions. It is also used to model the lifetime probability distributions of mechanical devices, since their failure rate functions are usually increasing with operating time.

Lower confidence limit estimation procedures for system reliability are needed during the development phase of systems to provide indications of a contractor's ability to meet a stated system reliability goal as development progresses and the results of test programs become available. These procedures are also needed to assess the reliability of systems that have been operating in the field for some time and have accumulated histories of failure data and unique configurations of modified or repaired components.

Few textbooks on reliability treat the problem of system reliability interval estimation. Those that do usually limit the discussion to series or parallel systems. Moreover, the procedures they present are not adaptable to other more complex systems. *Mann, Shafer, and Singpurwalla* [Ref.1 pp 487-524] provide one of the better treatments of a variety of these methods in Chapter 10 of their book. This chapter provides an excellent summative discussion of the many procedures that were developed from 1954 to 1974. However, none of the procedures reviewed in their book can accomodate the use of test data from a mix of components with *both*

exponential and *Weibull* failure time distributions. The procedures presented in this thesis does accomodate this type of system with a mixture of different component types. In addition, the procedures presented in this thesis can accomodate an additional mix of components for which only *attribute* data has been collected.

Procedures developed in this thesis are extensions of a procedure developed by *Myhre, Sanders and Rosenfeld* [Ref.2]. In their paper, they assume the failure times of continuously operating components have exponential distributions with associated failure rates, λ_i. The test data on the remaining components, the number of observed failures f_i in n_i tests, are assumed to have Poisson distributions with associated means $n_i q_i$. They assume the ratios λ_i/λ_j, q_i/q_j and λ_i/q_j are known and develop confidence interval estimation procedures for system reliability that use this information. They also show that the accuracy of their procedure is not very sensitive to moderate inaccuracies of these ratios. This suggests that it might be possible to estimate the ratios from the data as part of the interval estimation process and not suffer significant loss of accuracy in the interval estimates. Estimating these ratios is part of the procedures developed in this thesis.

This thesis also provides an annotated computer program that can be used to assess the accuracies of the lower confidence limit procedures when applied to any specific system. Sufficient annotations are provided throughout the program in Appendix C. This program provides the user with a means for verifying the accuracy of these proposed lower confidence limit procedures for his specific system and testing program, that is, sample sizes and type of truncation. This capability will allow the user to answer many *"what if"* type of questions.

II. THEORY

A. Interval Estimation Procedure for Exponential Failure Times

A system is defined to be *quasi-coherent* if an increase in reliability of any one of its components does not cause a decrease in system reliability. The components of a quasi-coherent system do not need to be statistically independent. However, throughout this thesis, it is assumed that all components are *statistically independent*.

Suppose a quasi-coherent system has k components and the distribution of the failure time of component i is *exponential* with failure rate λ_i. Then the system reliability R_s can be written as a function of λ_i , $i = 1, 2, ..., k$ as follows:

$$R_s(t) = g(\ \lambda_1, \lambda_2, ..., \lambda_k,\ t_1, t_2, ..., t_k\ ) \qquad \text{... (2.1)}$$

where t_i is the operating time for component i. Let m be any one of the k components and $r_i = \lambda_i/\lambda_m$, for $i = 1, 2, ..., k$. Then equation (2.1) may be viewed as

$$R_s(t) = g(\ \lambda_m,\ r_1, r_2, ..., r_k,\ t_1, t_2, ..., t_k\ ) \qquad \text{... (2.2)}$$

If the r_i's are known and $\hat{\lambda}_{m,U(\alpha)}$ were an upper 100(1-α)% confidence limit for λ_m, the corresponding lower confidence limit for $R_s(t)$ would be:

$$\hat{R}_s(t)_{L(\alpha)} = g(\ \hat{\lambda}_{m,U(\alpha)},\ r_1, r_2, ..., r_k,\ t_1, t_2, ..., t_k\ ) \qquad \text{... (2.3)}$$

Specifically, if we have a *series* system of independent components, so that

$$\begin{aligned} R_s(t) &= \exp\{\ -\sum_{i=1}^{k} \lambda_i t_i\ \} \\ &= \exp\{\ -\lambda_m \sum_{i=1}^{k} r_i t_i\ \} \end{aligned} \qquad \text{... (2.4)}$$

then,

$$\hat{R}_s(t)_{L(\alpha)} = \exp\{ -\hat{\lambda}_{m,U(\alpha)} \sum_{i=1}^{k} r_i t_i \} \qquad \text{... (2.5)}$$

If n_i items of component i are tested until f_i failures occur, T_i denotes the total test time accumulated on all the n_i items, and $F = \sum_{i=1}^{k} f_i$ then the expression

$$2\lambda_m \sum_{i=1}^{k} r_i T_i$$

has a *Chi-square* distribution with $2F$ degrees of freedom. See *Bain and Engelhardt* [Ref.3]. The corresponding 100(1-α)% upper confidence limit for λ_m is

$$\hat{\lambda}_{m,U(\alpha)} = \frac{\chi^2_{\alpha,2F}}{2\sum_{i=1}^{k} r_i T_i} \qquad \text{... (2.6)}$$

where $\chi^2_{\alpha,2F}$ is the 100(1-α)th percentile point of a *Chi-square* distribution with $2F$ degrees of freedom.

If the testing on component i is terminated when a total test time of T_i has been accumulated by all n_i items, then the equation for $\hat{\lambda}_{m,U(\alpha)}$ becomes

$$\hat{\lambda}_{m,U(\alpha)} = \frac{\chi^2_{\alpha,2(1+F)}}{2\sum_{i=1}^{k} r_i T_i} \qquad \text{... (2.7)}$$

In this case f_i is random and so is F.

If testing on each of the n_i items of component i are tested until a planned test time or failure, and failed items are *replaced* immediately, then equation (2.7) will be the exact expression for $\hat{\lambda}_{m,U(\alpha)}$. If failures are *not replaced*, then equation (2.7) is approximate. See *Lee, Bain and Englehardt* [Ref.3 pp 486-495]. Department of

Defense document NAVSEA OD29304B "Reliability and Availability Evaluation Program Manual" [Ref.4 p 5-42] provides *nearly exact* procedures for $\hat{\lambda}_{m,U(\alpha)}$ when testing is terminated by planned test time for each item tested and failures are not replaced.

The values of the r_i's are assumed to be *unknown* in this thesis. When testing is terminated by the number of failures, a nearly unbiased estimator for r_i is

$$\hat{r}_i = \frac{\hat{\lambda}_i}{\hat{\lambda}_m} \qquad \text{... (2.8)}$$

where $\hat{\lambda}_i = (f_i - 1)/T_i$ and the index m denotes the component with largest value of $\hat{\lambda}_i$. The ratio, $(f_i\text{-}1)/T_i$, is an unbiased estimator for λ_i (see Appendix A). If $1/\hat{\lambda}_m$ were unbiased for $1/\lambda_m$ then $\hat{r}_i$ would be an unbiased estimator for r_i. Replacing $\hat{\lambda}_m$ with $\hat{\lambda}_m f_m/(f_m - 1)$ in equation (2.8) will yield an unbiased estimator $\hat{r}_i$ for r_i. Multiplying by this constant $f_m/(f_m\text{-}1)$ is nullified by a cancellation with the same constant in the final equation for the system reliability lower confidence limit, so equation (2.8) is used to estimate r_i. Using estimator $\hat{r}_i$ for r_i, equation (2.6) becomes

$$\hat{\lambda}_{m,U(\alpha)} = \frac{\chi^2_{\alpha,2F}}{2\sum_{i=1}^{k} \hat{r}_i T_i} \qquad \text{... (2.9)}$$

It is important to note that the index m denotes the component for which $\hat{\lambda}_i = (f_i\text{-}1)/T_i$ is the largest among all the components in the system. The corresponding equation for the $100(1\text{-}\alpha)\%$ lower confidence limit on the reliability of a series system is

$$\hat{R}_s(t)_{L(\alpha)} = \exp\{ -\hat{\lambda}_{m,U(\alpha)} \sum_{i=1}^{k} \hat{r}_i t_i \} \qquad \text{... (2.10)}$$

The corresponding lower confidence limit for the reliability of any quasi-coherent system is given by equation (2.3) with r_i replaced by $\hat{r}_i$.

In this thesis, equation (2.8) with $\hat{\lambda}_i = f_i/T_i$ will also be used to estimate r_i under *exponential* assumptions when testing is terminated after an accumulated test time is achieved *(truncated)*, and when at least two components have at least one failed test item. This is done because failures will not be replaced in the time truncated test plans that are simulated in this thesis. In this type of testing both f_i and T_i are random. Under time truncation, it is possible that no failures will occur on any component tested in which case equation (2.8) is undefined. Also, if only one component has one failure and the remaining components have zero failure, equation (2.8) would be zero for all i except the case when $i = m$.

All of the confidence limit procedures in this thesis have a common special method for computing the lower confidence limit of system reliability in the two cases of zero or one failure. This feature amounts to a modification to equations (2.3) and (2.10).

When either zero failures or one failure have occured among all components, the test data is examined for each component to determine the total number, N_i , of *equivalent* component mission tests (i = 1, 2, ..., k). These N_i's and the system configuration are analyzed to determine the *equivalent* number of mission tests, N, for the system that would have occurred if N_1 , N_2 , ..., N_k of these k components were assembled into systems.

For a series system, this N will be equivalent to $min\ \{ N_1 , N_2 , ..., N_k \}$. The 100(1-$\alpha$)% lower confidence limit of system reliability if zero failures occurred is then computed directly as follows:

$$\hat{R}_s(t)_{L(\alpha)} = \sqrt[N]{\alpha} \qquad \text{... (2.11)}$$

If exactly one failure occurred among all k components, $\hat{R}_s(t)_{L(\alpha)}$ will be the solution for p in the equation

$$p^N + Np^{N-1}(1-p) = \alpha \qquad \text{... (2.12)}$$

These confidence limit equations are the standard *binomial* lower confidence limit equations. Equations (2.11) and (2.12) are part of the set of equations used to compute $\hat{R}_s(t)_{L(\alpha)}$ for all of the time truncated interval estimation procedures in this thesis.

It is important to remember that the symbol T_i in equations (2.7), (2.8) and (2.9) denote total accumulated test time for component i ; that is

$$T_i = \sum_{j=1}^{n_i} T_{ij} \qquad \text{... (2.13)}$$

where T_{ij} is the test time accumulated on the jth test item of component i and n_i is the number of test items of component i being tested.

B. Interval Estimation Procedure for Weibull Failure Times

Consider a *series* system with k components. Let the time to failure, X_i , of component i have a *Weibull* distribution with density

$$f_i(t_i) = \lambda_i^{\beta_i} \beta t_i^{\beta_i - 1} \exp\{ -(\lambda_i t_i)^{\beta_i} \} , \quad t_i > 0 \qquad \text{... (2.14)}$$

Then

$$R_i(t_i) = \exp\{ -(\lambda_i t_i)^{\beta_i} \} , \quad t_i > 0 \qquad \text{... (2.15)}$$

and

$$
\begin{aligned}
R_s(t) &= \prod_{i=1}^{k} \exp\{-\lambda_i^{\beta_i} t_i^{\beta_i}\} \\
&= \exp\{-\sum_{i=1}^{k} \lambda_i^{\beta_i} t_i^{\beta_i}\} \qquad \text{... (2.16)} \\
&= \exp\{-\lambda_m^{*} \sum_{i=1}^{k} r_i t_i^{\beta_i}\}
\end{aligned}
$$

where $\lambda_i^* = \lambda_i^{\beta i}$, λ_m^* is any one of the λ_i^* , $i = 1, 2, ..., k'$, and $r_i = \lambda_i^*/\lambda_m^*$. If the β_i's are known, then $X_i^{\beta i}$ will have a constant failure rate $\lambda_i^{\beta i}$ and the procedures described in Section A can be used to obtain $\hat{R}_s(t)_{L(\alpha)}$ with T_{ij} replaced by $T_{ij}^{\beta i}$ in equation (2.13).

Suppose β_i is unknown and $X_{i(1)}$, $X_{i,(2)}$, ..., $X_{i(fi)}$ are the *ordered* failure times under either type of truncated testing for component i in the system. Solutions $\hat{\beta}_i$ *and* $\hat{\lambda}_i$ for β_i and λ_i in the two equations given in equation (2.17) are the *maximum likelihood estimates* for β_i and λ_i. See *Mann and others* [Ref.1 pp 189-191]. These equations are used for both types of test truncation. If for component i, testing is terminated on the f_i^{th} failure, then $t_{is} = X_{i(fi)}$ in equation (2.17). The solution, $\hat{\beta}_i$, is a biased estimator for β_i. *Bain* [Ref.5 pp 220] provides a table of constants $B(n_i)$ which depends on number of test items n_i such that $\hat{\beta}_i^* = \hat{\beta}_i B(n_i)$ is a nearly unbiased estimator for β_i.

$$
\frac{\sum_{j=1}^{f_i} X_{i(j)}^{\beta_i} \ln X_{i(j)} + (n_i - f_i) t_{is}^{\beta_i} \ln t_{is}}{\sum_{j=1}^{f_i} X_{i(j)}^{\beta_i} + (n_i - f_i) t_{is}^{\beta_i}} - \frac{1}{\beta_i} = \frac{1}{f_i} \sum_{j=1}^{f_i} \ln X_{i(j)} \qquad \text{... (2.17a)}
$$

and

$$\lambda_i^{\beta_i} = \frac{f_i}{\sum_{j=1}^{f_i} X_{i(j)}^{\beta_i} + (n_i - f_i) t_{is}^{\beta_i}} \qquad \text{... (2.17b)}$$

If the testing for component i is terminated at failure or at a given time t_{oi} for each of the n_i items on test, then $t_{is} = t_{oi}$ in equations (2.17a) and (2.17b).

Now, let

$$T_{ij} = X_{ij}^{\hat{\beta}_i} \ , \quad with \quad \begin{array}{l} i = 1, 2, ..., k \\ j = 1, 2, ..., n_i \end{array} \qquad \text{... (2.18)}$$

In this thesis, the distribution of T_{ij} is approximated by the *exponential* distribution with failure rate $\lambda_i^{\hat{\beta}_i} \equiv \lambda_i^*$ and procedures similar to those in Section A are used to obtain the lower confidence limit on system reliability. Define

$$\hat{\lambda}_i^* = \frac{f_i}{T_i} \qquad \text{... (2.19)}$$

where $T_i = \sum_{j=1}^{n_i} T_{ij}$, $i = 1, 2, ..., k$. Let $\hat{\lambda}_m^* = \max_{alli} \hat{\lambda}_i^*$.

Note that this defines the index m. Define $\hat{r}_i$ by

$$\hat{r}_i = \frac{\hat{\lambda}_1^*}{\hat{\lambda}_m^*} = \hat{\lambda}_i^* . (\frac{T_m}{f_m}) \qquad \text{... (2.20)}$$

for both types of test truncation plans.

Then an approximate 100(1-α)% upper confidence limit for λ_m^* is given by

$$\hat{\lambda}^*_{m,U(\alpha)} = \frac{\chi^2_{\alpha,2F^*}}{2\sum_{i=1}^{k} \hat{r}_i T_i} \qquad \text{... (2.21)}$$

where

$$F^* = \begin{cases} \sum_{i=1}^{k} f_i \;, & \textit{if test till } f_i^{\,th} \textit{ failure for all components} \\ 1 + \sum_{i=1}^{k} f_i \;, & \textit{if test till specified time for all components} \end{cases} \qquad \text{... (2.22)}$$

The corresponding approximate 100(1-α)% lower confidence limit $R_s(t)_{L(\alpha)}$ for the reliability $R_s(t)$ of a series system is given by

$$\hat{R}_s(t)_{L(\alpha)} = \exp\{ -\hat{\lambda}^*_{m,U(\alpha)} \sum_{i=1}^{k} \hat{r}_i t_i^{\hat{\beta}_i} \} \qquad \text{... (2.23)}$$

when at least two components have at least one failure. Equations (2.11) and (2.12) also apply here when the total failures over all components is either zero or one.

The accuracies of these approximate confidence interval procedures were evaluated by using computer simulations which are described in the next chapter. During this evaluation process, the degrees of freedom in the expressions $\chi^2_{\alpha,2F^*}$ in equation (2.21), $\chi^2_{\alpha,2F}$ in equation (2.9) and $\chi^2_{\alpha,2(1+F)}$ in equation (2.7) were increased and decreased from the defined values of F^* and F given by these equations. The purpose of these modifications was to find more accurate lower confidence limit procedures. The specific increases and decreases are described in Chapter III. The results show that for some cases the procedures with modified degrees of freedom are more accurate.

III. COMPUTER SIMULATION

A. Test Plan 1 : Testing n_i Until f_i Failures (RETP1)

RETP1 is a program written in FORTRAN, on the Amdahl mainframe computer, which performs the computer simulation of the random failure times of the different types of components in the system. A documentation of this program and its associated subroutines is included in Appendix B.

The program accepts input parameters via an input file IN1.DAT. For each replication, it generates the failure times for all the component items included in the test plan using a uniform random number generating subroutine LRNDPC. A quick evaluation of LRNDPC (see Appendix D) by plotting $U(n+1)$ *vs* $U(n)$ illustrates the uniformity of the routine. The program determines the total test time accumulated for each component in the system and computes the estimates of the key parameters and the consequent lower confidence limit for system reliability for that replication. The process is repeated 1000 times. When all replications are done, the routine EVAL processes the lower confidence limit estimates from all 1000 replications and determines the two measures of accuracy for the run, namely RSLOW and LEVEL.

RSLOW is the 100(1-α) percentile of the *ordered* set of lower confidence limits from the 1000 replications computed in a run. The true reliability of the system is RS. The closer RSLOW is to RS, the greater the accuracy of the procedure under evaluation in the run. If the procedure is exact, RSLOW will be equivalent to RS. To be conservative, RSLOW should always be lower than RS.

LEVEL measures the proportion of 1000 lower confidence limits, from a run with 1000 replications, which are *lower* than the true system reliability RS. The closer LEVEL is to the specified confidence level for the procedure, 1-α, the better the procedure. Values of LEVEL greater than (1-α) reflect an under-estimation of RS which is conservative. Values of LEVEL lesser than 1-α signal an over-estimation of RS which may be undesirable.

Simulation runs are performed using RETP1 for all combinations of failure time distributions and levels of key input parameters listed below.

(a) System.
- 8 Exponential components in Series (Case 1)
- 8 Weibull components in Series (Case 2)
- 4 Exp and 4 Wei (Mixed) components in Series (Case 3)

(b) True System Reliability (RS).

- Hi (greater than 0.9) (Type A)
- Lo (greater than 0.8) (Type B)

(c) Level of Significance (α).

- 0.1
- 0.2

(d) Degrees of Freedom for χ^2 statistic (DF) as a function of the total number of failed test components (NFC) and total number of system components (NCOMP).

- DF = 2 * NFC
- DF = 2 * (NFC + NCOMP)
- DF = 2 * (NFC - NCOMP)
- DF = 2 * NFC - NCOMP

e) Test Plan for each component.

- Test 5 until 5 failures
- Test 15 until 15 failures
- Test 15 until 11 failures
- Test 15 until 7 failures
- Test 15 until 3 failures

For the 8 *exponential* components in Case 1, the mission time for each of the component is chosen to be 10 hrs. The program will accomodate different component mission times. The chosen values of the scale parameters, λ_i , were different depending on whether the system is highly reliable (Type A) or one with a lower reliability (Type B). The ratios between the largest and the smallest failure rate was chosen to be 8 and 4.5 respectively for Type A and Type B systems.

For the 8 *Weibull* components in Case 2, the mission time for each of the components was chosen to be 10 hrs. The chosen values of the scale parameters, λ_i, were different depending on whether the system is highly reliable (Type A) or one with a lower reliability (Type B). The ratio between the largest and smallest failure rate was chosen to be 8 for both system types. The shape parameter is chosen to be 2 for all cases. The program will accomodate any value greater than zero for the shape parameter.

A mixture of *exponential* and *Weibull* components with those parameters described in the last two paragraphs is chosen for the Type A and Type B systems of Case 3.

Each simulation run of 1000 replication results in an output file OUT1.DAT. The raw output from all the RETP1 runs are summarized in tabular form and placed in Appendix E. Each table corresponds to a specific run case and system type combination.

B. Test Plan 2 : Testing for a Specified Planned Test Time (RETP2)

RETP2 is another program written in FORTRAN, on the Amdahl mainframe computer, which performs the computer simulation of the random failure times of the different types of components in the system. A documentation of this program and its associated subroutines is included in Appendix C.

The structure of this program is quite similar to that of RETP1 described in Section A of this chapter. The program accepts input parameters via an input file IN2.DAT. For each replication, it generates the failure times for all the component items included in the test plan using LRNDPC. The program then determines the number of failed test components for each component in the system and computes the estimates of the key parameters and the consequent lower confidence limit for system reliability for that replication. The process is repeated 1000 times. When all replications are done, the routine EVAL processes the lower confidence limit estimates from all the 1000 replications and determines the two measures of accuracy for the run, namely RSLOW and LEVEL. The definitions of these two measures were discussed in the Section A.

Simulation runs are performed using RETP2 for all combinations of failure time distributions and levels of key input parameters listed below.

(a) System.
- 8 Exponential components in Series (Case 4)
- 8 Weibull components in Series (Case 5)
- 4 Exp and 4 Wei (Mixed) components in Series (Case 6)

(b) True System Reliability (RS).

- Hi (greater than 0.9) (Type A)
- Lo (greater than 0.8) (Type B)

(c) Level of Significance (α).

- 0.1
- 0.2

(d) Degrees of Freedom for χ^2 statistic (DF) as a function of the total number of failed test components (NFC) and total number of system components (NCOMP).

- DF = 2 * (1 + NFC)
- DF = 1.3 * 2 * (1 + NFC)

(e) 10 values of K = 0.25, 0.5, 1, 2, 3, 4, 5, 10, 20, 30

where K is a factor such that the expected number of failures for an exponential component during the specified total test time for that component is 0.6 times K. This K factor determines the bounds of the *expected* total number of failed test items, E[NFC]. The accuracy of the lower confidence limit procedures are highly correlated with E[NFC].

For the 8 *exponential* components in Case 4, the mission time for each of the component is chosen to be 5 hrs. The program can accomodate different component mission times. The chosen values of the scale parameters, λ_i, were different depending on whether the resultant system is highly reliable (Type A) or one with a lower reliability (Type B). The failure rate was chosen to be 0.001 and 0.005 failures/hr for all the components respectively. Total test time to be accumulated by each component is computed according to the following method. For each component i , t_i represents the amount of operating time required to result in a 40% survival probability, that is, an expected failure of 0.6 component with an exponential failure time distribution. The computation for t_i is as follows:

$$\begin{aligned} R_i(t_i) &= \exp(-\lambda t_i) \\ &= 0.4 \\ \therefore\ t_i &= -(\frac{1}{\lambda})\ln(0.4) \end{aligned}$$

T_i , the total amount of test time to be accumulated for component i would be K times t_i which will result in an expected number of 0.6 times K failed items for this component. E[NFC] will then be 8 times that number, since there are 8 such components in the system.

For the 8 *Weibull* components in Case 5, the mission time for each of the component is chosen to be 15 hrs. The chosen values of the scale parameters, λ_i, were 0.005 failures/hr for the Type A system and 0.01 failures/hr for a Type B system. The shape parameter is chosen to be 2. For each *Weibull* component, a maximum of 20 test items are tested to failure. Estimation of E[NFC] and thus the total test time to be accumulated for each component is based on the exponential failure time model described in the earlier paragraphs.

A mixture of *exponential* and *Weibull* components with those parameters described in the last two paragraphs is chosen for the Type A and Type B systems of Case 6.

Each simulation run of 1000 replication results in an output file OUT2.DAT. The raw output from all the RETP2 runs are summarized in tabular form and placed in Appendix F. Each table corresponds to a specific run case and system type combination.

IV. RESULTS AND DISCUSSION

The results of the simulation runs are summarized and discussed in this section. Tables 1A, 1B, 2A, 2B, 3A and 3B in Appendix E, and Tables 4A, 4B, 5A, 5B, 6A and 6B in Appendix F present the accuracy results in tabular form for all run cases that were simulated: a few of these tables appear in this section to facilitate discussion of the results.

A. Test Plan 1 : Testing n_i Until f_i Failures (RETP1)

Table 1A displays the simulation results for Case 1 Type A. In this case, the system is comprised of 8 components in series. The failure time of each component has an *exponential* distribution. The failure rates of the 8 components range from 0.0002 failures/hour to 0.0016 failures/hour. The mission time of the system is 10 hours and the mission operating time of each component was also set to 10 hours. The component mission times do not need to be equal to the system mission time for the procedures evaluated in this thesis. This was discussed in Chapter II and is allowed for in all of the lower confidence limit equations. System reliability, RS, in Table 1A is 0.931 . Throughout this case, all of these parameters remain fixed.

In simulation number 1 (S/N: 1) in Table 1A, five items for each of the 8 components in the system are tested until they fail. Thus the number of failed components (NFC) is 40. One set of this 40 failure times is randomly generated for each simulation run. For this set of data, four 90% lower confidence limits and four 80% lower confidence limits are computed. The four limits correspond to the values assigned to the degrees of freedom parameter F in the symbol $\chi^2_{\alpha,2F}$ which is a factor in the upper confidence limit equation for $\hat{\lambda}_{U(\alpha)}$. The four different methods for computing this degree of freedom appear in the "Deg of Freedom" column. NCOMP denotes the number of components in the system. Thus for each simulation run, eight lower confidence limits are computed. After 1000 replications of this

Table 1A : 8 Exp in Series, RS = 0.931 (Hi)
min λ = 0.0002 f/hr, max λ = 0.0016 f/hr, UT = 10 hrs

S/N	Test Plan	Deg of Freedom	α	Measures of Accuracy	
				RSLOW	LEVEL
1	Test 5 until 5 failed. NFC=40	2*NFC (80)	0.1	0.919	0.982
			0.2	0.919	0.960
		2*(NFC+ NCOMP) (96)	0.1	0.906	1.000
			0.2	0.906	0.999
		2*NFC- NCOMP (72)	0.1	0.927	0.949
			0.2	0.929	0.880
		2*(NFC- NCOMP) (64)	0.1	0.934	0.8[illegible]
			0.2	0.933	0.702
2	Test 15 until 15 failed. NFC=120	2*NFC (240)	0.1	0.928	0.955
			0.2	0.927	0.908
		2*(NFC+ NCOMP) (256)	0.1	0.923	0.990
			0.2	0.923	0.975
		2*NFC- NCOMP (232)	0.1	0.930	0.916
			0.2	0.932	0.833
		2*(NFC- NCOMP) (224)	0.1	0.932	0.844
			0.2	0.932	0.747
3	Test 15 until 11 failed. NFC=88	2*NFC (176)	0.1	0.927	0.955
			0.2	0.926	0.916
		2*(NFC+ NCOMP) (192)	0.1	0.921	0.996
			0.2	0.920	0.988
		2*NFC- NCOMP (168)	0.1	0.930	0.916
			0.2	0.929	0.843
		2*(NFC- NCOMP) (160)	0.1	0.933	0.843
			0.2	0.932	0.735

Table 1A : 8 Exp in Series, RS = 0.931 (Hi) (Cont...)
min λ = 0.0002 f/hr, max λ = 0.0016 f/hr, UT = 10 hrs

S/N	Test Plan	Deg of Freedom	α	Measures of Accuracy	
				RSLOW	LEVEL
4	Test 15 until 7 failed. NFC=56	2*NFC (112)	0.1	0.924	0.970
			0.2	0.923	0.931
		2*(NFC+ NCOMP) (128)	0.1	0.915	0.998
			0.2	0.913	0.994
		2*NFC- NCOMP (104)	0.1	0.929	0.919
			0.2	0.928	0.753
		2*(NFC- NCOMP) (96)	0.1	0.934	0.835
			0.2	0.933	0.720
5	Test 15 until 3 failed. NFC=24	2*NFC (48)	0.1	0.915	0.975
			0.2	0.912	0.975
		2*(NFC+ NCOMP) (64)	0.1	0.891	1.000
			0.2	0.888	1.000
		2*NFC- NCOMP (40)	0.1	0.927	0.944
			0.2	0.926	0.860
		2*(NFC- NCOMP) (32)	0.1	0.939	0.753
			0.2	0.939	0.634

simulation are run, the 2 measures of accuracy RSLOW and LEVEL are computed. The lower confidence limit procedures are exact if RSLOW = RS in which case LEVEL = 1-α.

Table 1A displays the accuracy results for 5 different sampling plans which are described in S/N: 1, 2, 3, 4 and 5.

A comparison of the four values of RSLOW for each of these five sampling plans reveal that the lower confidence limit procedure with degrees of freedom equal

to 2*NFC-NCOMP is the most accurate lower confidence limit procedure. In S/N 1, for example, the RSLOW value of 0.927 is the largest such value below the RS value of 0.931. Values of RSLOW above RS are optimistic and not as desirable as values of RSLOW which are equi-distant below RS.

The values of RSLOW and LEVEL are based on 1000 replications and their accuracy merit should roughly be measured to the nearest one hundredth. That is we should round 0.927 to 0.93 and compare it with RS = 0.93. It is evident that the lower confidence limit procedure with degrees of freedom equal to 2*NFC-NCOMP is very accurate for all cases simulated.

Table 2A displays the accuracy results of Case 2 for Type A systems. In this case, the 8 components connected in series have the shape parameter $\beta = 2$ and the scale parameters λ_i varying between 0.001 and 0.008 failures/hr. Mission time is 10 hours and each component has this same mission or utilization time (UT). Inspection of Table 2A reveals the following:

(1) More than 5 items of each component should be tested until failure for any of these procedures to be reasonably accurate.

(2) If 15 items of each component are tested until all fail, then these procedures will be reasonably accurate when the degrees of freedom is either 2*NFC-NCOMP or 2*(NFC-NCOMP).

(3) The procedures are reasonably accurate for 80% confidence level when the truncation is not below 7 out of 15 items.

(4) The procedures are slightly conservative at the 90% confidence level when the truncation is 7 out of 15 items or 11 out of 15 items.

There are numerous ways to modify these lower confidence limit procedures to effect improvements in their accuracy. One avenue is to modify the estimate for the shape parameter, β. Some very recent work in the literature provides a method for estimating β that differs greatly from the maximum likelihood estimator (MLE) and does not require computer iteration.

Table 2A : 8 Wei in Series, RS = 0.980 (Hi)
min λ = 0.001 f/hr, max λ = 0.008 f/hr, UT = 10 hrs

S/N	Test Plan	Deg of Freedom	α	Measures of Accuracy	
				RSLOW	LEVEL
1	Test 5 until 5 failed. NFC=40	2*NFC (80)	0.1	0.947	0.992
			0.2	0.930	0.989
		2*(NFC+ NCOMP) (96)	0.1	0.937	0.994
			0.2	0.918	0.993
		2*NFC- NCOMP (72)	0.1	0.951	0.989
			0.2	0.937	0.986
		2*(NFC- NCOMP) (64)	0.1	0.956	0.985
			0.2	0.983	0.981
2	Test 15 until 15 failed. NFC=120	2*NFC (240)	0.1	0.978	0.918
			0.2	0.974	0.913
		2*(NFC+ NCOMP) (256)	0.1	0.977	0.931
			0.2	0.978	0.924
		2*NFC- NCOMP (232)	0.1	0.979	0.914
			0.2	0.975	0.901
		2*(NFC- NCOMP) (224)	0.1	0.980	0.904
			0.2	0.975	0.889
3	Test 15 until 11 failed. NFC=88	2*NFC (176)	0.1	0.982	0.876
			0.2	0.977	0.860
		2*(NFC+ NCOMP) (192)	0.1	0.980	0.981
			0.2	0.979	0.882
		2*NFC- NCOMP (168)	0.1	0.983	0.861
			0.2	0.978	0.989
		2*(NFC- NCOMP) (160)	0.1	0.983	0.840
			0.2	0.979	0.819

Table 2A : 8 Wei in Series, RS = 0.980 (Hi) (Cont...)
min λ = 0.001 f/hr, max λ = 0.008 f/hr, UT = 10 hrs

S/N	Test Plan	Deg of Freedom	α	Measures of Accuracy	
				RSLOW	LEVEL
4	Test 15 until 7 failed. NFC=56	2*NFC (112)	0.1	0.987	0.800
			0.2	0.981	0.779
		2*(NFC+ NCOMP) (128)	0.1	0.985	0.839
			0.2	0.978	0.824
		2*NFC- NCOMP (104)	0.1	0.988	0.776
			0.2	0.982	0.753
		2*(NFC- NCOMP) (96)	0.1	0.989	0.746
			0.2	0.983	0.732
5	Test 15 until 3 failed. NFC=24	2*NFC (48)	0.1	0.994	0.621
			0.2	0.991	0.584
		2*(NFC+ NCOMP) (64)	0.1	0.993	0.705
			0.2	0.988	0.685
		2*NFC- NCOMP (40)	0.1	0.995	0.548
			0.2	0.992	0.514
		2*(NFC- NCOMP) (32)	0.1	0.996	0.468
			0.2	0.993	0.417

Table 3A displays the results of Case 3 for Type A systems. In this case 8 components are connected in series. Four of them have failure times with *exponential* distributions and the remaining four have failure times with *Weibull* distributions each with shape parameter, β, equal to 2.

Table 3A : 4 Exp and 4 Wei (Mixed) in Series, RS = 0.980 (Hi)
min λ = 0.002 f/hr, max λ = 0.008 f/hr, UT = 10 hrs

S/N	Test Plan	Deg of Freedom	α	Measures of Accuracy	
				RSLOW	LEVEL
1	Test 5 until 5 failed. NFC=40	2*NFC (80)	0.1	0.979	0.942
			0.2	0.978	0.905
		2*(NFC+ NCOMP) (96)	0.1	0.975	0.987
			0.2	0.974	0.976
		2*NFC- NCOMP (72)	0.1	0.981	0.881
			0.2	0.980	0.805
		2*(NFC- NCOMP) (64)	0.1	0.983	0.771
			0.2	0.982	0.684
2	Test 15 until 15 failed. NFC=120	2*NFC (240)	0.1	0.981	0.863
			0.2	0.980	0.800
		2*(NFC+ NCOMP) (256)	0.2	0.979	0.941
			0.2	0.979	0.898
		2*NFC- NCOMP (232)	0.1	0.981	0.881
			0.2	0.980	0.898
		2*(NFC- NCOMP) (224)	0.1	0.98[illegible]	0.725
			0.2	0.981	0.631
3	Test 15 until 11 failed. NFC=88	2*NFC (176)	0.1	0.98[illegible]	0.864
			0.2	0.980	0.801
		2*(NFC+ NCOMP) (192)	0.1	0.974	0.951
			0.2	0.978	0.907
		2*NFC- NCOMP (168)	0.1	0.982	0.802
			0.2	0.981	0.898
		2*(NFC- NCOMP) (160)	0.1	0.982	0.702
			0.2	0.982	0.591

Table 3A : 4 Exp and 4 Wei (Mixed) in Series, RS = 0.980 (Hi) (Cont...)
min λ = 0.002 f/hr, max λ = 0.008 f/hr, UT = 10 hrs

S/N	Test Plan	Deg of Freedom	α	Measures of Accuracy	
				RSLOW	LEVEL
4	Test 15 until 7 failed. NFC=56	2*NFC (112)	0.1	0.981	0.865
			0.2	0.980	0.787
		2*(NFC+ NCOMP) (128)	0.1	0.982	0.952
			0.2	0.978	0.920
		2*NFC- NCOMP (104)	0.1	0.982	0.769
			0.2	0.982	0.676
		2*(NFC- NCOMP) (96)	0.1	0.983	0.644
			0.2	0.983	0.523
5	Test 15 until 3 failed. NFC=24	2*NFC (48)	0.1	0.982	0.843
			0.2	0.981	0.762
		2*(NFC+ NCOMP) (64)	0.1	0.976	0.970
			0.2	0.975	0.941
		2*NFC- NCOMP (40)	0.1	0.984	0.684
			0.2	0.984	0.580
		2*(NFC- NCOMP) (32)	0.1	0.987	0.459
			0.2	0.987	0.356

Inspection of Table 3A reveals the following:

(1) The two procedures corresponding to degrees of freedom equal to 2*NFC and 2*NFC-NCOMP are reasonably accurate for all 5 simulation cases.

(2) The procedures appears to be nearly equally accurate for both 80% and 90% confidence levels.

B. Test Plan 2 : Testing for a Specified Planned Test Time (RETP2)

In the simulations for test plan 2, components were tested until failure or until some planned test time scenario. Failed items were not replaced. Components whose failure times had *exponential* distributions were tested until a pre-determined total test time was accumulated for their type of component. Components whose failure times had *Weibull* distributions were tested until failure or a pre-determined planned test time for that test item. The latter truncation plan is needed for *Weibull-type* items in order to use the maximum likelihood estimates [as in equation (2.17)] to solve for $\hat{\beta}$.

Inspection of Table 4A reveals that the lower confidence limit procedure for degrees of freedom equal to 2*(1+NFC) is quite accurate when enough testing is done to make the expected number of failures, E[NFC], greater than or equal to 4.8. This testing constraint is well within the domain of constraints set on testing in development programs for major systems within the Department of Defense.

Examination of Table 5A reveals that the lower confidence limit procedure for degrees of freedom equal to 2*(1+NFC) is moderately accurate when enough testing is done to make E[NFC] greater than or equal to 9. The accuracy diminishes slightly as E[NFC] increases. This could be corrected by decreasing the degrees of freedom slightly to make RSLOW slightly larger.

The results displayed in Table 6A show that the lower confidence limit procedure for degrees of freedom equal to 2*(1+NFC) is quite accurate when enough testing is done to make E[NFC] greater than or equal to 9.6.

Table 4A : 8 Exp in Series, RS = 0.961 (Hi)
λ = 0.001 f/hr, UT = 5 hrs

S/N	Degrees of Freedom	K / E[NFC] (TT)	α	Measures of Accuracy	
				RSLOW	LEVEL
1	2*(1+NFC)	0.25 / 1.2 (225)	0.1	0.957	0.851
			0.1	0.935	0.851
		0.5 / 2.4 (450)	0.1	0.957	0.857
			0.2	0.954	0.857
		1.0 / 4.8 (900)	0.1	0.957	0.914
			0.2	0.957	0.850
		2.0 / 9.6 (1800)	0.1	0.958	0.916
			0.2	0.960	0.850
		3.0 / 14.4 (2700)	0.1	0.959	0.916
			0.2	0.959	0.809
		4.0 / 19.2 (3600)	0.1	0.959	0.937
			0.2	0.960	0.843
		5.0 / 24 (4500)	0.1	0.960	0.926
			0.2	0.960	0.814
		10.0 / 48 (9000)	0.1	0.960	0.924
			0.2	0.960	0.809
		20.0 / 96 (18000)	0.1	0.960	0.914
			0.2	0.961	0.820
		30.0 / 144 (27000)	0.1	0.961	0.906
			0.2	0.961	0.804

Table 5A : 8 Wei in Series, RS = 0.956 (Hi) (*)
λ = 0.005 f/hr, UT = 15 hrs

S/N	Degrees of Freedom	K / E[NFC] (TT)	α	Measures of Accuracy	
				RSLOW	LEVEL
1	2*(1+NFC)	0.25 / 1.2 (45)	0.1	1.000	0.186
			0.2	1.000	0.158
		0.5 / 2.4 (90)	0.1	0.980	0.501
			0.2	0.979	0.458
		1.0 / 4.8 (180)	0.1	0.967	0.767
			0.2	0.960	0.732
		2.0 / 9.6 (360)	0.1	0.957	0.879
			0.2	0.952	0.854
		3.0 / 14.4 (540)	0.1	0.952	0.934
			0.2	0.946	0.922
		4.0 / 19.2 (720)	0.1	0.952	0.940
			0.2	0.946	0.928
		5.0 / 24 (900)	0.1	0.952	0.940
			0.2	0.946	0.928
		10.0 / 48 (1800)	0.1	0.952	0.940
			0.2	0.946	0.928
		20.0 / 96 (3600)	0.1	0.952	0.940
			0.2	0.946	0.928
		30.0 / 144 (5400)	0.1	0.952	0.940
			0.2	0.946	0.928

(*) 20 test items for each *Weibull* component.

Table 6A : 4 Exp and 4 Wei (Mixed) in Series, RS = 0.958 (Hi) (*)
λ(exp) = 0.001 f/hr, UT(exp) = 5 hrs
λ(wei) = 0.005 f/hr, UT(wei) = 15 hrs

S/N	Degrees of Freedom	K / E[NFC] TT(exp) TT(wei)	α	Measures of Accuracy	
				RSLOW	LEVEL
1	2*(1+NFC)	0.25 / 1.2 (225) (45)	0.1	1.000	0.684
			0.2	0.995	0.451
		0.5 / 2.4 (450) (90)	0.1	0.957	0.803
			0.2	0.975	0.874
		1.0 / 4.8 (900) (180)	0.1	0.971	0.736
			0.2	0.965	0.684
		2.0 / 9.6 (1800) (360)	0.1	0.964	0.803
			0.2	0.960	0.765
		3.0 / 14.4 (2700) (540)	0.1	0.964	0.874
			0.2	0.956	0.634
		4.0 / 19.2 (3600) (720)	0.1	0.960	0.873
			0.2	0.957	0.839
		5.0 / 24 (4500) (900)	0.1	0.956	0.887
			0.2	0.956	0.861
		10.0 / 48 (9000) (1800)	0.1	0.956	0.891
			0.2	0.956	0.862
		20.0 / 96 (18000) (3600)	0.1	0.956	0.892
			0.2	0.956	0.867
		30.0 / 144 (27000) (5400)	0.1	0.960	0.877
			0.2	0.956	0.862

The accuracy results of simulations performed in this thesis *cannot* be extended to systems that differ significantly from those simulated here. However, it can be said that the procedures which are accurate for series systems are also usually accurate for 1-out-of-k parallel systems because system reliability, R_s, can be written in terms of component reliabilities, R_i, as

$$R_s = 1 - \prod_{i=1}^{k} (1 - R_i)$$

Thus, upper confidence limits on $\prod_{i=1}^{k} (1 - R_i)$ will yield a lower confidence limit on R_s. The accuracy of the upper confidence interval procedures for $\prod_{i=1}^{k} (1 - R_i)$, a series-type problem, should be nearly the same as those obtained in this thesis because equations like (2.9) would be replaced with equations for the lower confidence limit $\hat{\lambda}_{m,L(\alpha)}$ on λ_m and would look like

$$\hat{\lambda}_{m,L(\alpha)} = \frac{\chi^2_{1-\alpha,2F}}{2\sum_{i=1}^{k} \hat{r}_i T_i} \qquad \text{... (4.1)}$$

If the degrees of freedom parameter F is large, as it is in the cases simulated in this thesis, the associated *Chi-Square* distribution with $2F$ degrees of freedom is nearly symmetric about its mean so the lower tail has nearly the same shape as the upper tail. This characteristic should yield very similar accuracies for $\hat{\lambda}_{m,L(\alpha)}$ as was obtained for $\hat{\lambda}_{m,U(\alpha)}$ in this thesis. These accuracy comparisons translate directly to similar comparisons about the accuracies of the associated lower confidence limit procedures for the system reliability of series and parallel systems.

Table 7 : Summary of Procedure Accuracy by Simulation Category (RETP1)

S/N	Key Parameters and their Levels	Observation and Discussion based on RSLOW and LEVEL
1	Run Cases ▪ All Exponential (1) ▪ All Weibull (2) ▪ Mixed (3)	▪ For all exponential systems, accurate procedures were developed for component sample sizes ≥ 5. ▪ For all Weibull systems, component sample sizes should be ≥ 15 with truncation at $r \geq 7$ failures.
2	System Reliabilities ▪ Hi (> 0.9) (Type A) ▪ Lo (> 0.8) (Type B)	Accurate procedures were developed for both cases if sample sizes are adequate.
3	Levels of Significance ▪ α = 0.1 ▪ α = 0.2	Accuracy varied slightly depending on system type and test plan, but accurate procedures exist for both levels.
4	Degrees of Freedom ▪ DF = 2*NFC ▪ DF = 2*(NFC+NCOMP) ▪ DF = 2*(NFC-NCOMP) ▪ DF = 2*NFC-NCOMP	Greatest accuracy as follows: ▪ All exponential : 2*NFC-NCOMP ▪ All Weibull : 2*NFC-NCOMP or 2*(NFC-NCOMP)
5	Test Plan ▪ Test 5 until 5 failures ▪ Test 15 until 15 failures ▪ Test 15 until 11 failures ▪ Test 15 until 7 failures ▪ Test 15 until 3 failures	Accurate procedures existed for all run cases for all system types except for the all Weibull system where number of test items $n \geq 15$ and number of failures r should be ≥ 7.

Table 8 : Summary of Procedure Accuracy by Simulation Category (RETP2)

S/N	Key Parameters and their Levels	Observation and Discussion based on RSLOW and LEVEL
1	Run Cases ▪ All Exponential (4) ▪ All Weibull (5) ▪ Mixed (6)	Procedures were accurate for DF = 2*(1+NFC) when enough testing was done to make the expected number of failed components E[NFC] ≥ 9.
2	System Reliabilities ▪ Hi (> 0.9) (Type A) ▪ Lo (> 0.8) (Type B)	Same as S/N(1).
3	Levels of Significance ▪ $\alpha = 0.1$ ▪ $\alpha = 0.2$	Same as S/N(1).
4	Degrees of Freedom ▪ DF = 2*(1+NFC) ▪ DF = 1.3*2*(1+NFC)	DF = 2*(1+NFC) was the most accurate procedure.
5	Test Plan ▪ K factors of 0.25, 0.5, 1, 2, 3, 4, 5, 10, 20, 30	K should be chosen so that E[NFC] ≥ 9.

Tables 7 and 8 provide cursory summaries of some constraints needed to assure the existence of one or more accurate lower confidence limit procedures among the procedures that were evaluated. The simulation scenarios are divided into five categories for this summarization.

V. APPLICATION EXAMPLES

Based on the procedures evaluated by the RETP1 and RETP2 runs, four different test plans and failure time data were constructed to illustrate the use of the procedures in providing a lower 100(1-α) % confidence limit for the system reliability of a series system with different types of components.

CASE 1 : 8 Exponential Components in Series
------- TEST PLAN 1 - Test 15 until 7 fails for each component

I. Raw Data

Comp i	T(1)	T(2)	T(3)	T(4)	T(5)	T(6)	T(7)
	Ordered Failure Times (h)						
1	300.0	400.0	500.0	600.0	700.0	800.0	900.0
2	350.0	450.0	550.0	650.0	750.0	850.0	950.0
3	400.0	500.0	600.0	700.0	800.0	900.0	1000.0
4	450.0	550.0	650.0	750.0	850.0	950.0	1050.0
5	500.0	600.0	700.0	800.0	900.0	1000.0	1100.0
6	550.0	650.0	750.0	850.0	950.0	1050.0	1150.0
7	600.0	700.0	800.0	900.0	1000.0	1100.0	1200.0
8	650.0	750.0	850.0	950.0	1050.0	1150.0	1250.0

II. Data Summary

Comp i	UT(i)	NC(i)	NF(i)	TT(i)	ELM(i)	ER(i)	ER(i)* TT(i)
1	5.0	15	7	11400.0	0.00053	1.00000	11400.0
2	5.0	15	7	12150.0	0.00049	0.93827	11400.0
3	5.0	15	7	12900.0	0.00047	0.88372	11400.0
4	5.0	15	7	13650.0	0.00044	0.83516	11400.0
5	5.0	15	7	14400.0	0.00042	0.79167	11400.0
6	5.0	15	7	15150.0	0.00040	0.75248	11400.0
7	5.0	15	7	15900.0	0.00038	0.71698	11400.0
8	5.0	15	7	16650.0	0.00036	0.68468	11400.0

III. Estimation Procedure for RSLOW

Parameter	df	Value	
ALPHA		0.1	
NFC		56	
CHISQD	112	131.56	(df = 2 * NFC , CHISQD from tables)
LMU		0.00072	
RSLOW		0.97647	

CASE 2 : 8 Weibull Components in Series
------- TEST PLAN 1 - Test 15 until 7 fails for each component

I. Raw Data

Comp i	Ordered Failure Times (h) T(1)	T(2)	T(3)	T(4)	T(5)	T(6)	T(7)
1	10.0	20.0	30.0	40.0	50.0	60.0	70.0
2	20.0	30.0	40.0	50.0	60.0	70.0	80.0
3	30.0	40.0	50.0	60.0	70.0	80.0	90.0
4	40.0	50.0	60.0	70.0	80.0	90.0	100.0
5	50.0	60.0	70.0	80.0	90.0	100.0	110.0
6	60.0	70.0	80.0	90.0	100.0	110.0	120.0
7	70.0	80.0	90.0	100.0	110.0	120.0	130.0
8	80.0	90.0	100.0	110.0	120.0	130.0	140.0

II. Data Summary

Comp i	UT(i)	NC(i)	NF(i)	TT(i)	ELM(i)	ER(i)	ER(i)* TT(i)
1	5.0	15	7	5.3E+04	1.32E-04	1.00000	5.3E+04
2	5.0	15	7	7.2E+04	9.79E-05	0.74406	5.3E+04
3	5.0	15	7	9.3E+04	7.54E-05	0.57328	5.3E+04
4	5.0	15	7	1.2E+05	5.98E-05	0.45431	5.3E+04
5	5.0	15	7	1.4E+05	4.85E-05	0.36842	5.3E+04
6	5.0	15	7	1.7E+05	4.01E-05	0.30452	5.3E+04
7	5.0	15	7	2.1E+05	3.37E-05	0.25577	5.3E+04
8	5.0	15	7	2.4E+05	2.87E-05	0.21777	5.3E+04

III. Estimation Procedure for RSLOW

Parameter	df	Value	
BETA		2.0	
ALPHA		0.1	
NFC		56	
CHISQD	112	131.56	(df = 2 * NFC , CHISQD from tables)
LMU		1.55E-04	
RSLOW		0.98497	

IV. Workarea

Comp i	Ordered Failure Times (h) raised to the power of BETA T'(1)	T'(2)	T'(3)	T'(4)	T'(5)	T'(6)	T'(7)
1	1.0E+02	4.0E+02	9.0E+02	1.6E+03	2.5E+03	3.6E+03	4.9E+03
2	4.0E+02	9.0E+02	1.6E+03	2.5E+03	3.6E+03	4.9E+03	6.4E+03
3	9.0E+02	1.6E+03	2.5E+03	3.6E+03	4.9E+03	6.4E+03	8.1E+03
4	1.6E+03	2.5E+03	3.6E+03	4.9E+03	6.4E+03	8.1E+03	1.0E+04
5	2.5E+03	3.6E+03	4.9E+03	6.4E+03	8.1E+03	1.0E+04	1.2E+04
6	3.6E+03	4.9E+03	6.4E+03	8.1E+03	1.0E+04	1.2E+04	1.4E+04
7	4.9E+03	6.4E+03	8.1E+03	1.0E+04	1.2E+04	1.4E+04	1.7E+04
8	6.4E+03	8.1E+03	1.0E+04	1.2E+04	1.4E+04	1.7E+04	2.0E+04

CASE 3 : 8 Exponential Components in Series
------- TEST PLAN 2 - Test until TT(i) for each component

I. Raw Data

No data except noting the number of failures for each component (NF(i))

II. Data Summary

Comp i	UT(i)	NF(i)	TT(i)	ELM(i)	ER(i)	ER(i)* TT(i)	ER(i)* UT(i)
1	5.0	6	2000.0	0.00300	0.85714	1714.3	4.3
2	5.0	6	2000.0	0.00300	0.85714	1714.3	4.3
3	5.0	5	2000.0	0.00250	0.71429	1428.6	3.6
4	5.0	7	2000.0	0.00350	1.00000	2000.0	5.0
5	5.0	5	2000.0	0.00250	0.71429	1428.6	3.6
6	5.0	5	2000.0	0.00250	0.71429	1428.6	3.6
7	5.0	4	2000.0	0.00200	0.57143	1142.9	2.9
8	5.0	7	2000.0	0.00350	1.00000	2000.0	5.0

III. Estimation Procedure for RSLOW

Parameter	df	Value	
ALPHA		0.1	
NFC		45	
CHISQD	92	109.76	(df = 2 * (1 + NFC) , CHISQD from tables)
LMU		0.00427	
RSLOW		0.87180	

CASE 4 : 8 Weibull Components in Series
------- TEST PLAN 2 - Test 20 until TT(i) for each component

I. Raw Data

Comp i	TT(i)	Ordered Failure Times (h) T(1)	T(2)	T(3)	T(4)	T(5)	T(6)
1	60.0	10.0	20.0	30.0	40.0	50.0	60.0
2	60.0	15.0	25.0	35.0	45.0	55.0	
3	60.0	20.0	30.0	40.0	50.0	60.0	
4	60.0	25.0	35.0	45.0	55.0		
5	60.0	30.0	40.0	50.0	60.0		
6	60.0	35.0	45.0	55.0			
7	60.0	40.0	50.0	60.0			
8	60.0	45.0	55.0				

II. Data Summary

Comp i	UT(i)	NC(i)	NF(i)	ET(i)	ELM(i)	ER(i)	ER(i)* ET(i)
1	5.0	20	6	6.0E+04	1.01E-04	1.00000	6.0E+04
2	5.0	20	5	6.1E+04	8.18E-05	0.81118	5.0E+04
3	5.0	20	5	6.3E+04	7.94E-05	0.78704	5.0E+04
4	5.0	20	4	6.5E+04	6.20E-05	0.61499	4.0E+04
5	5.0	20	4	6.6E+04	6.04E-05	0.59919	4.0E+04`
6	5.0	20	3	6.7E+04	4.45E-05	0.44090	3.0E+04
7	5.0	20	3	6.9E+04	4.35E-05	0.43179	3.0E+04
8	5.0	20	2	7.0E+04	2.86E-05	0.28394	2.0E+04

III. Estimation Procedure for RSLOW

Parameter	df	Value	
BETA		2.0	
ALPHA		0.1	
NFC		32	
CHISQD	66	81.09	(df = 2 * (1 + NFC) , CHISQD from tables)
LMU		1.28E-04	
RSLOW		0.98425	

IV. Workarea

Comp i	TT'(1)	Ordered Failure Times (h) raised to the power of BETA T'(1)	T'(2)	T'(3)	T'(4)	T'(5)	T'(6)
1	3.6E+03	1.0E+02	4.0E+02	9.0E+02	1.6E+03	2.5E+03	3.6E+03
2	3.6E+03	2.3E+02	6.3E+02	1.2E+03	2.0E+03	3.0E+03	
3	3.6E+03	4.0E+02	9.0E+02	1.6E+03	2.5E+03	3.6E+03	
4	3.6E+03	6.3E+02	1.2E+03	2.0E+03	3.0E+03		
5	3.6E+03	9.0E+02	1.6E+03	2.5E+03	3.6E+03		
6	3.6E+03	1.2E+03	2.0E+03	3.0E+03			
7	3.6E+03	1.6E+03	2.5E+03	3.6E+03			
8	3.6E+03	2.0E+03	3.0E+03				

VI. CONCLUSION

Some of the lower confidence limit procedures developed and evaluated in this thesis are reasonably accurate for the series systems simulated for test plans with sample sizes and truncation scenarios that are usually experienced in DoD aquisition programs. The accuracy of these methods can be varied by modifying the degrees of freedom parameter, F, in the $\chi^2_{\alpha,F}$ term in equation (2.9), namely:

$$\hat{\lambda}_{m,U(\alpha)} = \frac{\chi^2_{\alpha,2F}}{2\sum_{i=1}^{k} \hat{r}_i T_i}$$

The computer program can be modified with modest effort to accomodate specific complex coherent systems so long as the components have failure time distributions that are *exponential* or *Weibull*. This means that the computer program provided in this thesis can be used to develop a reasonably accurate lower confidence limit for the system reliability of a specific complex quasi-coherent system with *independent* components. This can be done by choosing the failure distribution and associated parameters of the components, the corresponding test plan parameters and the desired level of confidence. The simulation can then be run for this set of parameters for various equations for the degrees of freedom parameter, F, to determine an equation for F that yields a lower confidence limit with a satisfactory degree of accuracy.

When testing is truncated on the number of failures, a reasonably accurate procedure existed for all 3 cases of systems (all *exponential*, all *Weibull* and *mixed*) that were simulated when (a) the sample size of the components was 10 or larger, and, (b) the ratio of the number of failures to the sample size was at least 0.5. When testing was truncated by planned test time, reasonably accurate procedures were found for cases where the expected number of failures was at least 7.

VII. RECOMMENDATIONS

The computer program developed in this thesis facilitates the development of lower confidence limit procedures for explicit quasi-coherent systems. Systems other than series systems with large numbers of components (eg. 30) should be simulated to test the versatility of the general lower confidence limit methods used here.

Modified estimates for the β (shape) parameter in the *Weibull* failure time distribution and the parameter $r = \lambda_i/\lambda_m$ should be explored in an attempt to find more accurate procedures.

APPENDIX A : Derivation of Formula Used

Suppose T has an *exponential* distribution with failure rate λ. Suppose $T_{(1)}, T_{(2)}, T_{(3)}, \ldots, T_{(r)}$ are the first r ordered statistics in a random sample of size n from this *exponential* distribution. Let S be defined by

$$S = \sum_{i=1}^{r} T_{(i)} + (n-r)T_{(r)}$$

It is well known that $2\lambda S$ has a *Chi-Square* distribution with $2r$ degrees of freedom (See Ref.3 p 488). The maximum likelihood estimator for λ is given by

$$\hat{\lambda} = \frac{r}{S}$$

We seek an unbiased estimator for $\hat{\lambda}$. Suppose X has a *Chi-Square* distribution with $2r$ degrees of freedom, then

$$f_x(x;r) = \frac{1}{2^r\Gamma(r)} x^{r-1} \exp(-\frac{x}{2})$$

and the integral of this function from zero to infinity equals 1 (See Ref.3).

$$\begin{aligned} E(\frac{1}{X}) &= \int_0^\infty \frac{1}{2^r\Gamma(r)} x^{(r-1)-1} \exp(-\frac{x}{2}) dx \\ &= \frac{\Gamma(r-1)}{2\Gamma(r)} \int_0^\infty \frac{1}{2^{r-1}\Gamma(r-1)} x^{(r-1)-1} \exp(-\frac{x}{2}) dx \\ &= \frac{1}{2(r-1)} \end{aligned}$$

Then

$$E(\hat{\lambda}) = E(\frac{r}{S}) = r2\lambda E(\frac{1}{2\lambda S}) = \frac{r\lambda}{(r-1)}$$

Therefore

$$E(\frac{r-1}{r}\hat{\lambda}) = E(\frac{r-1}{S}) = \lambda$$

APPENDIX B : Users' Guide for RETP1

Reliability Estimation Test Plan 1 (RETP1).
by YEE, Kah-Chee
July 91

1. Brief Description.

RETP1 is a computer program written in FORTRAN that runs on the Amdahl mainframe at NPGS. It allows the user to simulate exponential and Weibull failure times of component items being tested to evaluate the accuracy of a confidence limit estimation procedure based on Type II data censoring (that is, testing n_i items of component i until f_i of them failed).

2. Program Input. (IN1.DATA)

The input of the program are specified to the program via an input file called IN1.DAT. A sample input file is shown below.

```
This file contains the inputs required by the RETPl model.
Update only the numerical values between dotted lines as appropriate.
Do not delete any of the comment lines.   (IN1.DAT)  14 May 91
C---------------------------------------------------------------------
C Value       Type   Units  Description                        Variable
C---------------------------------------------------------------------
  16807.0     Real     -    initial random seed                  ISEED
  8           Int      -    total # of components in system      NCOMP
  4           Int      -    # of EXPonential components          NEXP
  4           Int      -    # of WEIbull components              NWEI
  0           Int      -    # of GEOmetric components            NGEO
  0.01        Real     -    tolerance for MLE                    TOL
  0.20        REAL     -    DESIRED SIGNIFICANCE LEVEL           ALPHA
  1000        Int      -    # of replications desired            NREP
  3           Int      -    test case number                     TCN
                              1 = all EXP
                              2 = all WEI
                              3 = EXP + WEI
                              4 = EXP + WEI + GEO
  8           Int      -    number of cut sets                   NCS
C---------------------------------------------------------------------
```

```
C TEST PLAN : Testing NC(I) items of component i
C             until NF(I) of them fails. (Use REAL numbers ONLY!!!)
C------------------------------------------------------------------------
C Comp    Comp    Comp Parameters         Util          Test Plan Inputs
C Number  Type    Scale       Shape     Time/Cycle      # Comp    # Failed
C  I      TY(I)  PARM1(I)    PARM2(I)  UT(I)  UC(I)    NC(I)     NF(I)
C Int     Int    Real        Real      (hrs)   Int      Int       Int
C------------------------------------------------------------------------
  1.0     1.0    0.0020       1.0      10.0             15.0        3.0
  2.0     1.0    0.0040       1.0      10.0             15.0        3.0
  3.0     1.0    0.0060       1.0      10.0             15.0        3.0
  4.0     1.0    0.0080       1.0      10.0             15.0        3.0
  5.0     2.0    0.0020       2.0      10.0             15.0        3.0
  6.0     2.0    0.0040       2.0      10.0             15.0        3.0
  7.0     2.0    0.0060       2.0      10.0             15.0        3.0
  8.0     2.0    0.0080       2.0      10.0             15.0        3.0
C-------------------------------------------------------------------------
C  Note :  TY(I)=1  EXPONENTIAL  P(surv) = exp(-PARM1)*T)
C          TY(I)=2  WEIBULL      P(surv) = exp(-(PARM1*T)**PARM2)
C          TY(I)=3  GEOMETRIC    P(surv) = PARM1**N
C-------------------------------------------------------------------------
C SYSTEM CONFIGURATION : Identification of CUT SETs
C                        - min groups of components that have to fail
C                          for the system to fail.
C-------------------------------------------------------------------------
C Cut Set    # in Set     List of Components in Cutset
C    J       COMP(J,1)    COMP(J,2) ... up to COMP(J,1) components
C-------------------------------------------------------------------------
     1           1        1  0  0  0  0  0  0  0  0
     2           1        2  0  0  0  0  0  0  0  0
     3           1        3  0  0  0  0  0  0  0  0
     4           1        4  0  0  0  0  0  0  0  0
     5           1        5  0  0  0  0  0  0  0  0
     6           1        6  0  0  0  0  0  0  0  0
     7           1        7  0  0  0  0  0  0  0  0
     8           1        8  0  0  0  0  0  0  0  0
C-------------------------------------------------------------------------
```

3. Program Flow and Logic. (NAME1.DEF, PARM1.DEF and RETP1.FOR)

Input parameters are first read in by the program by calling the INPUT subroutine. The program then evoke the SIM subroutine which generates the random failure times and compute the key statistics required in the procedure. The next subroutine EVAL determines the measures of accuracy for run. REPORT is the subroutine which generates the output file for the run OUT1.DAT.

The variables in the program RETP1.FOR are described in the file NAME1.DEF as listed below.

```
C-----------------------------------------------------------------------
C  This file contains the declaration for input and output variables
C  used in the the RETP1 model.  (NAME1.DEF)  14 May 91
C-----------------------------------------------------------------------
C  Input Variables.
C  ----------------
C  ISEED = initial random seed selected.
C  SEED = current random seed.
C  RS = true overall series system reliability.
C  ALPHA = level of significance desired.
C  NREP = number of replications desired for the simulation.
C  TPN = test plan number (1).
C  TCN = test case number (1, 2, 3 or 4).
C  NCOMP = total number of components in the system.
C  NEXP = number of components with EXP failure times.
C  NWEI = number of components with WEI failure times.
C  NGEO = number of components with GEO failure times.
C  TOL = desired tolerance for MLE of WEI shape parameter.
C  Distribution:    EXPonential       WEIbull       GEOmetric
C  TY(I) = type:        1                2             3
C     PARM(1,I):     Scale(1/hr)     Scale(1/hr)     Prob
C     PARM(2,I):        -              Shape           -
C  UT(I) = utilization time (hrs) for component i (EXP and WEI).
C  UC(I) = utilization cycles for component i (GEO only).
C  NC(I) = number of test samples (sample size) for cmponent i.
C  NF(I) = desired number of failures in test for component i.
C  NCS = number of cut-sets for the system.
C  COMP(J,K) = kth parameter of cut-set j (first being the no. of
C              components belonging to the cut-set).
C
C  Assumed Variables.
C  -----------------
C  MAXCOMP = maximum number of components allowed in the system.
C  MAXREP = maximum number of replications permitted.
C  MAXCUT = maximum number of cut-sets.
C
C  Program and Output Variables.
C  ---------------------------
C  RS = true overall system reliability.
C  TT(I) = total accumulated failure time (hr) for component i
C          (EXP and WEI only).
C  TC(I) = total accumulated cycles to failure (incl. failure cycle)
C          for component i (GEO only).
C  EBETA(I) = estimate for shape parameter of component i (if Weibull).
C  REL1(J) = actual reliability for cut-set j.
C  REL2(J) = computed reliability for cut-set j for current replication.
C  ELM(I) = estimated component failure rate (1/hrs) for component i.
C  ELMAX(M) = max estimated component failure rate for rep m (1/hrs).
C  ER(I) = ratio of estimated failure rate to LMAX.
C  NFC(M) = total number of failed test components.
```

```
C  LMU(M) = upper confidence bound for failure rate (1/hrs),
C  RSL(M) = lower confidence limit estimated for system reliability
C           ... for the mth replication.
C  ORSL(M) = ordered RSL(M) (ascending).
C  RSLOW = (1-ALPHA)x100 percentile of set of RSL(M).
C  LEVEL = achieved confidence level, ie. proportion of RSL(M) that
C          are lesser than RS (conservative estimate).
C
C-- END OF NAME1.DEF ----------------------------------------------
C
```

Together with the main program in RETP1.FOR are the other subroutines needed in the simulation. The declaration of variables is done in the file PARM1.DEF. Relevant descriptions are included as comment lines in the source code to help explain the program segments. A listing of PARM1.DEF and RETP1.FOR is given below.

```
C-----------------------------------------------------------------------
C  This file contains the declaration for input and output variables
C  used in the the RETP1 model.   (PARM1.DEF)  14 May 91
C-----------------------------------------------------------------------
      INTEGER MAXCOMP, MAXREP
      PARAMETER( MAXCOMP = 100 , MAXREP = 1000 , MAXCUT = 20 )
      REAL*8 ISEED, SEED
      INTEGER NREP, TCN, NCOMP, NEXP, NWEI, NGEO, NCS,
     *        NC(MAXCOMP), NF(MAXCOMP), TY(MAXCOMP), NFC(MAXREP),
     *        UC(MAXCOMP), TC(MAXCOMP), COMP(MAXCUT,MAXCOMP)
      REAL*8 RS, ALPHA, UT(MAXCOMP), TT(MAXCOMP),
     *       PARM(2,MAXCOMP), ELM(MAXCOMP), ER(MAXCOMP),
     *       LMU(MAXREP), RSL(MAXREP), ORSL(MAXREP),
     *       ELMAX(MAXREP), RSLOW, LEVEL, TOL, EBETA(MAXCOMP),
     *       REL1(MAXCUT), REL2(MAXCUT)
C
      COMMON/BLOCK1/ISEED, SEED, NREP, TCN, NCOMP, NC, NF, NEXP, NWEI,
     *              NGEO, NCS, TY, NFC, UC, TC, COMP
      COMMON/BLOCK2/RS, ALPHA, UT, TT, PARM, ELM, ER, LMU,
     *              RSL, ORSL, ELMAX, RSLOW, LEVEL, TOL, EBETA,
     *              REL1, REL2
C
C-- END OF PARM1.DEF -----------------------------------------------
C
C-----------------------------------------------------------------------
C  This file contains the main program and the subroutines
C  for the Reliability Estimation Test Plan 1 (RETP1) model.
C  (RETP1.FOR) - runs on a IBM PC Compatible.
C
C  MAINFRAME VERSION...
C
C  Test Plan 1 : Testing NC(I) items for component i
```

```
C ----------   Until NF(I) of them fails.
C
C by Yee Kah-Chee  SMC 2802.
C 14 May 91.
C-----------------------------------------------------------------------
C 1.  Main Program (RETP1).
C-----------------------------------------------------------------------
      PROGRAM RETP1
C
C Include the declaration files.
C
      INCLUDE 'NAME1 DEF'
      INCLUDE 'PARM1 DEF'
C
C Read in input data.
C
      CALL INPUT
C
C Activate simulation.
C
      CALL SIM
C
C Process and evaluate output data.
C
      CALL EVAL
C
C Generate simulation report.
C
      CALL REPORT
C
      STOP
      END
C-----------------------------------------------------------------------
C 2.  Input Initialisation Subroutine (INPUT).
C-----------------------------------------------------------------------
      SUBROUTINE INPUT
C
C This subroutine reads in the inputs for the RETP1 model.
C
C Include the declaration file.
C
      INCLUDE 'PARM1 DEF'
C
      INTEGER I, J, K, DUM2(11)
      REAL*8 DUM1(7)
C
C Read data from 'IN1.DAT' designated as logic unit 1.
C
      OPEN(UNIT=1)
C
```

```
      READ(1,10)
   10 FORMAT(1X,/////)
      READ(1,*) ISEED
      READ(1,*) NCOMP
      READ(1,*) NEXP
      READ(1,*) NWEI
      READ(1,*) NGEO
      READ(1,*) TOL
      READ(1,*) ALPHA
      READ(1,*) NREP
      READ(1,*) TCN
      READ(1,20)
   20 FORMAT(1X,///)
      READ(1,*) NCS
C
      READ(1,30)
   30 FORMAT(1X,////////)
C
      DO 50 K = 1, NCOMP
        READ(1,*) DUM1
C
        I = NINT(DUM1(1))
        TY(I) = NINT(DUM1(2))
C
        IF (TY(I).EQ.1) THEN
          PARM(1,I) = DUM1(3)
          PARM(2,I) = DUM1(4)
          UT(I) = DUM1(5)
          NC(I) = NINT(DUM1(6))
          NF(I) = NINT(DUM1(7))
          EBETA(I) = 0
        ELSEIF (TY(I).EQ.2) THEN
          PARM(1,I) = DUM1(3)
          PARM(2,I) = DUM1(4)
          UT(I) = DUM1(5)
          NC(I) = NINT(DUM1(6))
          NF(I) = NINT(DUM1(7))
        ELSEIF (TY(I).EQ.3) THEN
          PARM(1,I) = DUM1(3)
          PARM(2,I) = DUM1(4)
          UC(I) = NINT(DUM1(5))
          UT(I) = DUM1(5)
          NC(I) = NINT(DUM1(6))
          NF(I) = NINT(DUM1(7))
        ENDIF
   50   CONTINUE
C
      READ(1,60)
   60 FORMAT(1X,///////////)
      DO 80 I = 1, NCS
```

```
         READ(1,*) DUM2
         J = DUM2(1)
         COMP(J,1) = DUM2(2)
         DO 70 K = 1, COMP(J,1)
           COMP(J,K+1) = DUM2(K+2)
   70    CONTINUE
   80 CONTINUE
      CLOSE(UNIT=1)
      RETURN
      END
C-----------------------------------------------------------------------
C  3.  Subroutine (SIM)
C-----------------------------------------------------------------------
      SUBROUTINE SIM
C
C  This subroutine simulates NREP possible outcomes of the test plan
C  desired in order to obtain the raw estimates of LMU(M) and RSL(M)
C  for each of the replication.
C
C  Include the declaration file
C  and declare local variables.
C
      INCLUDE 'PARM1 DEF'
      INTEGER I, J, K, M, ISUM, KEY
      REAL*8 UNI
      REAL*8 SUM, PROD,
     *       FT(MAXCOMP), OFT(MAXCOMP)
C
      SEED = ISEED
C
C  Compute overall true system reliability RS.
C
      RS = 1.0
      DO 30 J = 1, NCS
        PROD = 1.0
        DO 20 I = 1, COMP(J,1)
          K = COMP(J,I+1)
          PROD = PROD*( 1 - SURV(TY(K),PARM(1,K),PARM(2,K),UT(K)) )
   20   CONTINUE
        REL1(J) = 1.0 - PROD
        RS = RS * REL1(J)
   30 CONTINUE
C
C  Start of Simulation.
C  (Initialize replication counter M).
C
      M = 1
      DO WHILE (M.LE.NREP)
C
C      PRINT 35, M
```

```
C    35 FORMAT(1X,'Replication ',I4)
C
C   Test Plan : Sample and determine unknown TT(I)
C   ---------   with known NC(I) until NF(I) fails.
C
C   Generate NC(I) failure times, put them in ascending order
C   with the smallest failure time on the top of the list.
C
          DO 70 I = 1, NCOMP
C
            DO 40 K = 1, NC(I)
              CALL LRNDPC(SEED,UNI,1)
C
              IF(TY(I).EQ.1) THEN
                FT(K) = -LOG(UNI)/PARM(1,I)
              ELSEIF(TY(I).EQ.2) THEN
                FT(K) = (1.0/PARM(1,I))*(-LOG(UNI))**(1.0/PARM(2,I))
              ELSEIF(TY(I).EQ.3) THEN
                FT(K) = 1.0
                DO WHILE (UNI.LT.PARM(1,I))
                  FT(K) = FT(K) + 1.0
                  CALL LRNDPC(SEED,UNI,1)
                ENDDO
              ENDIF
   40       CONTINUE
C
C   Bubble Sort the failure times in ascending order.
C
            CALL BUBBLE(NC(I),FT,OFT)
C
C   Compute the total time accumulated in the test and the estimate
C   for the failure rate of the component as in the procedure.
C
            IF (TY(I).NE.2) THEN
              SUM = 0.0
              DO 50 K = 1, NF(I)
                SUM = SUM + OFT(K)
   50         CONTINUE
              TT(I) = FLOAT(NC(I)-NF(I))*OFT(NF(I)) + SUM
              IF (TY(I).EQ.1) THEN
                ELM(I) = FLOAT(NF(I)-1)/TT(I)
              ELSEIF (TY(I).EQ.3) THEN
                ELM(I) = FLOAT(NF(I)-1)/TT(I)
              ENDIF
            ELSEIF (TY(I).EQ.2) THEN
C
C      PRINT 55, M, I
C   55 FORMAT(1X,'REP = ',I3, '     Comp = ',I3,/)
C
              CALL MLESHAPE(OFT,NC(I),NF(I),TOL,1.D0,EBETA(I))
```

```
C
          EBETA(I) = BN(NC(I))*EBETA(I)
C
          SUM = 0.0
          DO 60 K = 1, NF(I)
            SUM = SUM + OFT(K)**EBETA(I)
   60     CONTINUE
C
          TT(I) = FLOAT(NC(I)-NF(I))*OFT(NF(I))**EBETA(I) + SUM
          ELM(I) = FLOAT(NF(I))/TT(I)
        ENDIF
C
   70 CONTINUE
C
C  Determine the total number of failed test items.
C
      ISUM = 0
      DO 80 I = 1, NCOMP
        ISUM = ISUM + NF(I)
   80 CONTINUE
      NFC(M) = ISUM
C
C  Determine the maximum failure rate estimate
C  and identify that component.
C
      ELMAX(M) = 0.0
      KEY = 0
      DO 90 I = 1, NCOMP
        IF (ELM(I).GT.ELMAX(M)) THEN
          ELMAX(M) = ELM(I)
          KEY = I
        ENDIF
   90 CONTINUE
C
C  Compute the ratios of the failure rate estimates to their maximum.
C
      DO 100 I = 1, NCOMP
        ER(I) = ELM(I) / ELMAX(M)
  100 CONTINUE
C
C  Determination of LMU(M)
C
      SUM = 0.0
      DO 110 I = 1, NCOMP
          SUM = SUM + (ER(I)*TT(I))
  110 CONTINUE
C
      LMU(M) = CHISQD(1-ALPHA,2*(NFC(M)-NCOMP))/(2*SUM)
C
C  Compute estimate of overall reliability RSL(M) for the system.
```

```
C
      RSL(M) = 1.0
      DO 130 J = 1, NCS
        PROD = 1.0
        DO 120 I = 1, COMP(J,1)
          K = COMP(J,I+1)
          IF (TY(K).EQ.1) THEN
            PROD = PROD*(1 - SURV(TY(K),LMU(M)*ER(K),EBETA(K),UT(K)))
          ELSEIF (TY(K).EQ.2) THEN
            PROD = PROD*(1 - SURV(TY(K),(LMU(M)*ER(K))**(1./EBETA(K)),
     *                   EBETA(K),UT(K)))
          ELSEIF (TY(K).EQ.3) THEN
            PROD = PROD*(1 - SURV(TY(K),1.D0-LMU(M)*ER(K),0.D0,UT(K)))
          ENDIF
  120   CONTINUE
        REL2(J) = 1.0 - PROD
        RSL(M) = RSL(M) * REL2(J)
  130 CONTINUE
C
C  Increment replication counter.
C
        M = M + 1
C
      ENDDO
      END
C-----------------------------------------------------------------------
C  4.  Subroutine (EVAL).
C-----------------------------------------------------------------------
      SUBROUTINE EVAL
C
C  This subroutine calls BUBBLE to sort the array RSL(NREP) in
C  ascending order to get an ordered array ORSL(NREP).  It also
C  determine the estimate for RSLOW at the specified significance
C  level ALPHA and the value of LEVEL in which ORSL(LEVEL) is closest
C  to the true reliability RS.
C
C  Include the declaration files
C  and declare the local variables.
C
      INCLUDE 'PARM1 DEF'
      INTEGER INDEX
      REAL*8 DIFF
C
C  Order the array RSL(NREP) in ascending order.
C
      DO 10 M = 1, NREP
        ORSL(M) = RSL(M)
   10 CONTINUE
C
C  Bubble Sort.  Sink the larger of the pair.
```

```
C
      CALL BUBBLE(NREP,RSL,ORSL)
C
C  Determine the (1-ALPHA)% lower confidence bound for the system
C  reliability.
C
      RSLOW = ORSL(NINT(NREP*(1-ALPHA)))
C
C  Finding the % confidence level for the true reliability RS.
C  (ie. the proportion of RSL(M) lesser than RS)
C
      DIFF = 1.0
      INDEX = 0
      DO 200 M = 1, NREP
        IF (ABS(ORSL(M)-RS).LT.DIFF) THEN
          DIFF = ABS(ORSL(M)-RS)
          INDEX = M
        ENDIF
  200 CONTINUE
C
      LEVEL = FLOAT(INDEX)/NREP
C
C  Record evaluated parameters in RAW1.DAT (unit 2).
C
      OPEN(UNIT=2)
      WRITE(2,300)
  300 FORMAT(1X, '     M       LMU(M)    ELMAX(M)       RSL(M)',
     *          '     ORSL(M)    NFC(M)')
      DO 500 M = 1, NREP
        WRITE(2,400) M,LMU(M),ELMAX(M),RSL(M),ORSL(M),NFC(M)
  400   FORMAT(1X,I6,2F12.7,2F12.7,I10)
  500 CONTINUE
      CLOSE(UNIT=2)
C
      RETURN
      END
C-----------------------------------------------------------------------
C  5.  Report Generation Subroutine (REPORT).
C-----------------------------------------------------------------------
       SUBROUTINE REPORT
C
C  This subroutine record the simulation results into the 'OUT1.DAT'
C  file as logic unit 3.
C
C  Include the declaration files
C  and declare local variables.
C
      INCLUDE 'PARM1 DEF'
      INTEGER I, J, K, DUM(10)
C
```

```
C  Write to output file 'OUT1.DAT' designated as logic unit 3.
C
      OPEN(UNIT=3)
C
      WRITE(3,10)
      WRITE(3,20) NREP
      WRITE(3,25)
      WRITE(3,26)
      WRITE(3,30)
      WRITE(3,40)
      WRITE(3,50) ISEED,NCOMP,ALPHA,TOL,NCS,TCN
C
      WRITE(3,60)
      DO 200 I = 1, NCOMP
        WRITE(3,70) I,TY(I),PARM(1,I),PARM(2,I),UT(I),NC(I),NF(I)
  200 CONTINUE
      WRITE(3,80)
      WRITE(3,90)
      DO 300 I = 1, NCOMP
        WRITE(3,100) I,NF(I),TT(I),ELM(I),ER(I),EBETA(I)
  300 CONTINUE
      WRITE(3,110)
      WRITE(3,120)
      DO 500 J = 1, NCS
        DO 400 K = 1, 10
          DUM(K) = COMP(J,K)
  400   CONTINUE
        WRITE(3,130) J,DUM,REL1(J),REL2(J)
  500 CONTINUE
      WRITE(3,140)
      WRITE(3,150) RS,ELMAX(NREP),LMU(NREP),RSLOW,LEVEL
C
   10 FORMAT(1X,'OUT1.DAT : Output File of the RETP1 simulation')
   20 FORMAT(1X,'              after ',I5,' replications',/)
   25 FORMAT(1X,'COMMENTS :  8 COMPONENTS IN SERIES            ')
   26 FORMAT(1X,'             DF = 2 * (NFC - NCOMP)  ',/)
   30 FORMAT(1X,'Input Parameters:',/)
   40 FORMAT(1X,'     ISEED   NCOMP   ALPHA     TOL   NCS   TCN',/)
   50 FORMAT(1X,F10.1,I8,F8.4,F8.5,2I6,/)
   60 FORMAT(1X,'  I TY(I) PARM1(I) PARM2(I)   UT(I)   NC(I)   NF(I)',/)
   70 FORMAT(1X,I3,I6,2F9.5,F8.2,2I8)
   80 FORMAT(1X,/,'Output Parameters for the LAST Replication:',/)
   90 FORMAT(1X,'  I NF(I)           TT(I)         ELM(I)         ER(I)',
     *          '       EBETA(I)',/)
  100 FORMAT(1X,I3,I6,E16.7,2F14.7,F14.7)
  110 FORMAT(1X,/,'Cut-Set Data:',/)
  120 FORMAT(1X,'  J  NUM    Component List          ',
     *          '        REL1       REL2(M)',/)
  130 FORMAT(1X,I3,I5,9I3,2F12.9)
  140 FORMAT(1X,/,'            RS     ELMAX(M)      LMU(M)',
```

```
     *              '        RSLOW        LEVEL',/)
 150  FORMAT(1X,5F12.7,/)
C
      CLOSE(UNIT=3)
C
      RETURN
      END
C-----------------------------------------------------------------------
C  This portion of the file contains functions and subroutines
C  used in the RETP1 model.
C        - 14 May 91
C        - by Yee Kah-Chee SMC 2802
C-----------------------------------------------------------------------
C  A.  Random Number Generating Subroutine (LRNDPC).
C      (Courtesy of Mr. David Lim Hung-Heng)
C-----------------------------------------------------------------------
      SUBROUTINE LRNDPC (DSEED,U,N)
      INTEGER            N, I
      REAL*8             U(N)
      REAL*8             D31M1, DSEED, D31
C     D31M1=2**31 - 1
C     D31  =2**31
      DATA D31M1/2147483647.D0/
      DATA D31  /2147483648.D0/
      DO 5 I=1,N
C        DSEED = DMOD(950706376.D0*DSEED,D31M1)
         DSEED = DMOD(16807.D0*DSEED,D31M1)
    5 U(I) = DSEED / D31
      RETURN
      END
C-----------------------------------------------------------------------
C  B.  Survivability Function.
C-----------------------------------------------------------------------
      FUNCTION SURV(TYPE,PAR1,PAR2,UTIL)
C
C  This function returns the survival probability of the component of
C  different types (TYPE) with scale (PAR1) and shape (PAR2) parameters
C  given the specified utilization times or cycles (UTIL).
C
      INTEGER TYPE, N
      REAL*8 PAR1, PAR2, UTIL
C
      IF (TYPE.EQ.1) THEN
        SURV = EXP(-(PAR1*UTIL))
      ELSEIF (TYPE.EQ.2) THEN
        SURV = EXP(-((PAR1*UTIL)**PAR2))
      ELSE
        N = NINT(UTIL)
        SURV = PAR1**N
      ENDIF
```

```
C
      END
C-------------------------------------------------------------------------
C  C.  Bubble Sort Routine in ASCENDING Order.
C-------------------------------------------------------------------------
      SUBROUTINE BUBBLE(N,LIST,OLIST)
C
C  This subroutine performs a bubble sort in increasing order (ie. sink
C  the greater numeral) for the first N terms in an array LIST and
C  returns the result in OLIST.
C
      LOGICAL DONE
      INTEGER N, K, PAIR
      REAL*8 LIST(*), OLIST(*)
C
C  Sink the larger of the pair.
C
      DO 50 K = 1, N
        OLIST(K) = LIST(K)
   50 CONTINUE
      PAIR = N - 1
      DONE = .FALSE.
      DO WHILE (.NOT.DONE)
        DONE = .TRUE.
        DO 100 K = 1, PAIR
          IF (OLIST(K).GT.OLIST(K+1)) THEN
            TEMP = OLIST(K)
            OLIST(K) = OLIST(K+1)
            OLIST(K+1) = TEMP
            DONE = .FALSE.
          ENDIF
  100   CONTINUE
        PAIR = PAIR - 1
      ENDDO
      END
C-------------------------------------------------------------------------
C  D.  Unbiasing Factor for Biased MLE for Weibull Shape Parameter.
C-------------------------------------------------------------------------
      FUNCTION BN(I)
C
C  This functon returns the value of the unbiased factor for the biased
C  maximum likelihood estimate of the shape parameter of a Weibull
C  distribution with a sample size of N.
C
      INTEGER I
      IF (I.LE.5) THEN
        BN = (I*0.699)/(5.0)
      ELSEIF (I.EQ.6) THEN
        BN = 0.752
      ELSEIF (I.EQ.7) THEN
```

```
          BN = 0.786
        ELSEIF (I.EQ.8) THEN
          BN = 0.82
        ELSEIF (I.EQ.9) THEN
          BN = 0.8395
        ELSEIF (I.EQ.10) THEN
          BN = 0.859
        ELSEIF (I.EQ.11) THEN
          BN = 0.871
        ELSEIF (I.EQ.12) THEN
          BN = 0.883
        ELSEIF (I.EQ.13) THEN
          BN = 0.892
        ELSEIF (I.EQ.14) THEN
          BN = 0.901
        ELSEIF (I.EQ.15) THEN
          BN = 0.9075
        ELSEIF (I.EQ.16) THEN
          BN = 0.914
        ELSEIF (I.EQ.17) THEN
          BN = 0.9185
        ELSEIF (I.EQ.18) THEN
          BN = 0.923
        ELSEIF (I.EQ.19) THEN
          BN = 0.927
        ELSEIF (I.EQ.20) THEN
          BN = 0.931
        ELSEIF (I.LE.25) THEN
          BN = 0.931+(I-20)*0.014/5.0
        ELSEIF (I.LE.30) THEN
          BN = 0.945+(I-25)*0.01/5.0
        ELSEIF (I.LE.40) THEN
          BN = 0.955+(I-30)*0.011/10.0
        ELSEIF (I.LE.60) THEN
          BN = 0.966+(I-40)*0.012/20.0
        ELSEIF (I.LE.80) THEN
          BN = 0.978+(I-60)*0.006/20.0
        ELSEIF (I.LE.100) THEN
          BN = 0.984+(I-80)*0.003/20.0
        ELSEIF (I.LE.120) THEN
          BN = 0.987+(I-100)*0.003/20.0
        ELSE
          BN = 1.0
        ENDIF
        RETURN
        END
C-------------------------------------------------------------------------
C  E.  Biased MLE of Weibull Shape Parameter.
C-------------------------------------------------------------------------
        SUBROUTINE MLESHAPE(T,N,R,DEL,B,BNEW)
```

```
C
C  This subroutine returns a biased estimator (BNEW) for a Weibull
C  shape parameter using the Newton-Raphson's Method of Successive
C  Approximation.  The data parameters consist of an ascending ordered
C  list of failure times (T), sample size (N), number of failed samples
C  (R), tolerance for convergence (DEL) and an initial estimate of the
C  shape parameter (B).
C
      LOGICAL DONE
      INTEGER N, R, I
      REAL*8 GFUNCT, GPRIME, B, BOLD, BNEW, T(*), DEL,
     *       TERM1, TERM2, TERM3, SUM1, SUM2, SUM3, SUM4, STEP
C
      BNEW = B
      DONE = .FALSE.
C
      DO WHILE (.NOT.DONE)
C
        DONE = .TRUE.
        TERM1 = FLOAT(N-R)*(T(R)**BNEW)
        TERM2 = FLOAT(N-R)*(T(R)**BNEW)*LOG(T(R))
        TERM3 = FLOAT(N-R)*(T(R)**BNEW)*LOG(T(R))*LOG(T(R))
        SUM1 = 0.0
        SUM2 = 0.0
        SUM3 = 0.0
        SUM4 = 0.0
C
        DO 50 I = 1, R
          SUM1 = SUM1 + T(I)**BNEW
          SUM2 = SUM2 + (T(I)**BNEW)*LOG(T(I))
          SUM3 = SUM3 + (T(I)**BNEW)*LOG(T(I))*LOG(T(I))
          SUM4 = SUM4 + LOG(T(I))
   50   CONTINUE
C
        GFUNCT = (SUM2+TERM2)/(SUM1+TERM1) - (1.0/BNEW)
     *           - (1.0/FLOAT(R))*SUM4
C
        GPRIME = (1.0/(SUM1+TERM1)**2)*( (SUM1+TERM1)*(SUM3+TERM3)
     *                                - (SUM2+TERM2)**2 )
     *           + (1.0/BNEW**2)
C
C     PRINT 60, GFUNCT,GPRIME,BNEW
C  60 FORMAT(1X, 'GFUNCT =',F8.3,' GPRIME =',F8.3,' BNEW =',F8.3)
C
C  Control magnitude of the marching step towards convergence
C  as no more than 0.1.
C
        IF ((GFUNCT.LT.0) .AND. (GPRIME.GT.0)) THEN
          STEP = VMAX(-.1D0,(GFUNCT/GPRIME))
        ELSEIF ((GFUNCT.GT.0) .AND. (GPRIME.LT.0)) THEN
```

```
          STEP = VMAX(-.1D0,(GFUNCT/GPRIME))
        ELSE
          STEP = VMIN(.1D0,(GFUNCT/GPRIME))
        ENDIF
C
        BOLD = BNEW
        BNEW = BNEW - STEP
C
C  Check for convergence of the MLE for the shape parameter B.
C
       IF (ABS(BOLD-BNEW).GT.DEL) THEN
          DONE = .FALSE.
        ENDIF
C
C  Avoid overflow error due to large MLE value caused by small
C  GPRIME (slope) as GFUNCT approaches to near zero.
C  Stop when magnitude of BNEW exceeds 7.
C
       IF (BNEW.GT.7.0) THEN
         BNEW = BOLD
         DONE = .TRUE.
       ENDIF
C
      ENDDO
      RETURN
      END
C-----------------------------------------------------------------------
C  E.  Chi-Square Quantile Function.
C-----------------------------------------------------------------------
      FUNCTION CHISQD(P,N)
C
C  Modified version of Algorithm 451 from Comunications of the ACM
C  Aug 1977 Vol.16 No.8 .
C
C  This function evaluates the quantile at the probability level P
C  (left tail area) for the Chi-square distribution with
C  N degrees of freedom.
C
      REAL*8 P
      REAL X
      INTEGER IF
      DIMENSION C(21), A(19)
      DATA C/ 1.565326E-3,
     *        1.060438E-3,
     *       -6.950356E-3,
     *       -1.323293E-2,
     *        2.277679E-2,
     *       -8.986007E-3,
     *       -1.513904E-2,
     *        2.530010E-3,
```

```
     *        -1.450117E-3,
     *         5.169654E-3,
     *        -1.153761E-2,
     *         1.128186E-2,
     *         2.607083E-2,
     *        -0.2237368,
     *         9.780499E-5,
     *        -8.426812E-4,
     *         3.125580E-3,
     *        -8.553069E-3,
     *         1.348028E-4,
     *         0.4713941,
     *         1.0000886 /
      DATA A/ 1.264616E-2,
     *        -1.425296E-2,
     *         1.400483E-2,
     *        -5.886090E-3,
     *        -1.091214E-2,
     *        -2.304527E-2,
     *         3.135411E-3,
     *        -2.728484E-4,
     *        -9.699681E-3,
     *         1.316872E-2,
     *         2.618914E-2,
     *        -0.2222222,
     *         5.406674E-5,
     *         3.483789E-5,
     *        -7.274761E-4,
     *         3.292181E-3,
     *        -8.729713E-3,
     *         0.4714045,
     *         1. /
      IF (N-2) 10, 20, 30
   10 CALL XFROMP(.5*(1.-P),X,IF)
      CHISQD = X
      CHISQD = CHISQD*CHISQD
      RETURN
   20 CHISQD = -2.*LOG(1.-P)
      RETURN
   30 F = N
      F1 = 1./F
      CALL XFROMP(P,X,IF)
      T  = X
      F2 = SQRT(F1)*T
      IF (N.GE.(2+INT(4.*ABS(T)))) GO TO 40
      CHISQD = ((((((C(1)*F2+C(2))*F2+C(3))*F2+C(4))*F2
     *         +C(5))*F2+C(6))*F2+C(7))*F1+((((((C(8)+C(9)*F2)*F2
     *         +C(10))*F2+C(11))*F2+C(12))*F2+C(13))*F2+C(14)))*F1+
     *         (((((C(15)*F2+C(16))*F2+C(17))*F2+C(18))*F2
     *         +C(19))*F2+C(20))*F2+C(21)
```

```
      GO TO 50
   40 CHISQD=(((A(1)+A(2)*F2)*F1+(((A(3)+A(4)*F2)*F2
     *       +A(5))*F2+A(6)))*F1+(((((A(7)+A(8)*F2)*F2+A(9))*F2
     *       +A(10))*F2+A(11))*F2+A(12)))*F1+(((((A(13)*F2
     *       +A(14))*F2+A(15))*F2+A(16))*F2+A(17))*F2*F2
     *       +A(18))*F2+A(19)
   50 CHISQD = CHISQD*CHISQD*CHISQD*F
      RETURN
      END
C-----------------------------------------------------------------------
C  F.  Standard Normal Variate Computation Subroutine.
C-----------------------------------------------------------------------
      SUBROUTINE XFROMP(P,X,IFAULT)
C
C  Algorithm AS 24 J.R.STAT.SOC. C. (1969) Vol.18. No.3.
C
C  This subroutine computes the standard normal deviate X for
C  the specified left tail area P.
C
      REAL*8 P
      DIMENSION A(5)
      DIMENSION CONNOR (17), HSTNGS(6)
      DATA CONNOR/ 8.0327350124E-17,
     *             1.4483264644E-15,
     *             2.4668270103E-14,
     *             3.9554295164E-13,
     *             5.9477940136E-12,
     *             8.3507027951E-11,
     *             1.0892221037E-9,
     *             1.3122532964E-8,
     *             1.4503852223E-7,
     *             1.4589169001E-6,
     *             1.3227513228E-5,
     *             1.0683760684E-4,
     *             7.5757575758E-4,
     *             4.6296296296E-3,
     *             2.3809523810E-2,
     *             0.1,
     *             0.33333333333 /
C
      DATA RTHFPI / 1.2533141373 /
C
      DATA RRT2PI / 0.3989422804 /
C
      DATA TERMIN / 1.0E-11 /
C
      DATA HSTNGS / 2.515517,
     *              0.802853,
     *              0.010328,
     *              1.432788,
```

```
     *              0.189269,
     *              0.001308 /
C
      IFAULT = 1
      IF ((P.LE.0.0).OR.(P.GE.1.0)) GO TO 100
      IFAULT = 0
C
C Get first approximation XO to deviate by Hastings' formula
C
      B = P
      IF(B.GT.0.5) B = 1.0 - B
C
      F = - LOG(B)
      E = SQRT(F+F)
      XO = -E + ((HSTNGS(3)*E+HSTNGS(2))*E+HSTNGS(1))/
     *     (((HSTNGS(6)*E+HSTNGS(5))*E+HSTNGS(4))*E+1.0)
      IF (XO.LT.0.0) GO TO 1
      XO = 0.0
      PO = 0.5
      X1 = -RTHFPI
      GO TO 7
C
C Find the area PO corresponding to XO
C
    1 Y = XO**2
      IF (XO.LE.-1.9) GO TO 3
      Y = -0.5*Y
C
C (1) series approximation
C
      PO = CONNOR(1)
      DO 2 L=2,17
    2 PO = PO*Y + CONNOR(L)
      PO = (PO*Y+1.0)*XO
      X1 = -(PO+RTHFPI)*EXP(-Y)
      PO = PO*RRT2PI + 0.5
      GO TO 7
C
C (2) continued fraction approximation
C
    3 Z = 1.0/Y
      A(2) = 1.0
      A(3) = 1.0
      A(4) = Z + 1.0
      A(5) = 1.0
      W = 2.0
C
    4 DO 6 L=1,3,2
      DO 5 J=1,2
      K = L + J
```

```
      KA = 7 - K
C
    5 A(K) = A(KA) + A(K)*W*Z
C
    6 W = W + 1.0
      APPRXU = A(2)/A(3)
      APPRXL = A(5)/A(4)
      C = APPRXU - APPRXL
      IF (C.GE.TERMIN) GO TO 4
      X1 = APPRXL/XO
      PO = -X1*RRT2PI*EXP(-0.5*Y)
C
C Get accurate value of deviate by Taylor Series
C (X1, X2, X3 are derivatives for the Taylor Series
C
    7 D = F + LOG(PO)
      X2 = XO*X1*X1 -X1
      X3 = X1**3 + 2.0*XO*X1*X2 -X2
      X = ((X3*D/3.0+X2)*D/2.0+X1)*D + XO
      IF (P.LE.0.5) GO TO 100
      X = -X
  100 RETURN
      END
C-----------------------------------------------------------------------
C  G.  Maximum Function.
C-----------------------------------------------------------------------
      FUNCTION VMAX(X,Y)
      REAL*8 X, Y
      IF(X.GT.Y) THEN
        VMAX = X
      ELSE
        VMAX = Y
      ENDIF
      RETURN
      END
C-----------------------------------------------------------------------
C  H.  Minimum Function.
C-----------------------------------------------------------------------
      FUNCTION VMIN(X,Y)
      REAL*8 X,Y
      IF(X.LT.Y) THEN
        VMIN = X
      ELSE
        VMIN = Y
      ENDIF
      RETURN
      END
```

4. Program Output. (OUT1.DAT)

The result for the simulation run based on the input parameters specified in IN1.DAT are computed and written to the file OUT1.DAT. A sample of this file is as follows.

```
OUT1.DAT : Output File of the RETP1 simulation
           after  1000 replications

COMMENTS :  8 COMPONENTS IN SERIES
            DF = 2 * (NFC - NCOMP)

Input Parameters:

     ISEED    NCOMP    ALPHA      TOL    NCS    TCN

   16807.0        8   0.2000 0.01000      8      3

   I TY(I) PARM1(I) PARM2(I)   UT(I)    NC(I)    NF(I)

   1     1  0.00200  1.00000   10.00       15        3
   2     1  0.00400  1.00000   10.00       15        3
   3     1  0.00600  1.00000   10.00       15        3
   4     1  0.00800  1.00000   10.00       15        3
   5     2  0.00200  2.00000   10.00       15        3
   6     2  0.00400  2.00000   10.00       15        3
   7     2  0.00600  2.00000   10.00       15        3
   8     2  0.00800  2.00000   10.00       15        3

Output Parameters for the LAST Replication:

   I NF(I)           TT(I)        ELM(I)          ER(I)        EBETA(I)

   1     3   0.1969091E+04    0.0010157      0.1703872      0.0000000
   2     3   0.3355078E+03    0.0059611      1.0000000      0.0000000
   3     3   0.5999981E+03    0.0033333      0.5591815      0.0000000
   4     3   0.6423594E+03    0.0031135      0.5223055      0.0000000
   5     3   0.2563836E+16    0.0000000      0.0000000      6.3524983
   6     3   0.8186629E+07    0.0000004      0.0000615      2.9571536
   7     3   0.5675037E+09    0.0000000      0.0000009      4.2418658
   8     3   0.5675519E+05    0.0000529      0.0088672      2.3471863

Cut-Set Data:

   J  NUM    Component List                  REL1       REL2(M)

   1    1  1  0  0  0  0  0  0  0  0 0.980198622 0.990279973
   2    1  2  0  0  0  0  0  0  0  0 0.960789382 0.944286704
   3    1  3  0  0  0  0  0  0  0  0 0.941764534 0.968452990
   4    1  4  0  0  0  0  0  0  0  0 0.923116326 0.970502377
   5    1  5  0  0  0  0  0  0  0  0 0.999600053 0.999999940
   6    1  6  0  0  0  0  0  0  0  0 0.998401225 0.999680758
```

7	1	7	0	0	0	0	0	0	0	0 0.996406436	0.999911249
8	1	8	0	0	0	0	0	0	0	0 0.993620396	0.988757312

RS	ELMAX(M)	LMU(M)	RSLOW	LEVEL
0.8089645	0.0059611	0.0057325	0.8691733	0.3380000

APPENDIX C : Users' Guide for RETP2

Reliability Estimation Test Plan 2 (RETP2).
by YEE, Kah-Chee
July 91

1. Brief Description.

RETP2 is a computer program written in FORTRAN that runs on the Amdahl mainframe at NPGS. It allows the user to simulate exponential and Weibull failure times of component items being tested to evaluate the accuracy of a confidence limit estimation procedure based on Type I data censoring (that is, testing items of component i until a specified total testing time is achieved for each of them).

2. Program Input. (IN2.DATA)

The input of the program are specified to the program via an input file called IN2.DAT. A sample input file is shown below.

```
This file contains the inputs required by the RETP2 model.
Update only the numerical values between dotted lines as appropriate.
Do not delete any of the comment lines.  (IN2.DAT)  20 Jun 91
C----------------------------------------------------------------------
C Value        Type   Units  Description                        Variable
C----------------------------------------------------------------------
  16807.0      Real     -    initial random seed               ISEED
  8            Int      -    total # of components in system   NCOMP
  4            INT      -    # OF EXPONENTIAL COMPONENTS       NEXP
  4            INT      -    # OF WEIBULL COMPONENTS           NWEI
  0            INT      -    # OF GEOMETRIC COMPONENTS         NGEO
  0.01         Real     -    tolerance for MLE                 TOL
  0.20         REAL     -    DESIRED SIGNIFICANCE LEVEL        ALPHA
  1000         INT      -    # OF REPLICATIONS DESIRED         NREP
  3            INT      -    TEST CASE NUMBER                  TCN
                               1 = all EXP
                               2 = all WEI
                               3 = EXP + WEI
                               4 = EXP + WEI + CYC
  8            Int      -    number of cut sets                NCS
C----------------------------------------------------------------------
```

```
C TEST PLAN : Testing until TT(I) (total test time) is accumulated.
C             (Use REAL numbers ONLY!!!)
C-----------------------------------------------------------------------
C Comp    Comp     Comp Parameters        Util      Test Plan Inputs
C Number  Type   Scale        Shape    Time/Cycle     Accumulated
C  I      TY(I)  PARM1(I)    PARM2(I)    UT(I)          TT(I)      NC(I)
C Int     Int    Real        Real     (hrs)(cycs) (hrs)(cycs)     Int
C-----------------------------------------------------------------------
  1.0     1.0    0.005        1.0          5.0       5400.0        20.0
  2.0     1.0    0.005        1.0          5.0       5400.0        20.0
  3.0     1.0    0.005        1.0          5.0       5400.0        20.0
  4.0     1.0    0.005        1.0          5.0       5400.0        20.0
  5.0     2.0    0.010        2.0         15.0       2700.0        20.0
  6.0     2.0    0.010        2.0         15.0       2700.0        20.0
  7.0     2.0    0.010        2.0         15.0       2700.0        20.0
  8.0     2.0    0.010        2.0         15.0       2700.0        20.0
C-----------------------------------------------------------------------
C  Note :  TY(I)=1  EXPONENTIAL  P(surv) = exp(-PARM2)*T)
C          TY(I)=2  WEIBULL      P(surv) = exp(-(PARM1*T)**PARM2)
C          TY(I)=3  GEOMETRIC    P(surv) = PARM1**T
C-----------------------------------------------------------------------
C SYSTEM CONFIGURATION : Identification of CUT SETs
C                        - min groups of components that have to fail
C                          for the system to fail.
C-----------------------------------------------------------------------
C Cut Set    # in Set     List of Components in Cutset
C    J       COMP(J,1)    COMP(J,2) ... up to COMP(J,1) components
C-----------------------------------------------------------------------
     1          1         1  0  0  0  0  0  0  0  0  0
     2          1         2  0  0  0  0  0  0  0  0  0
     3          1         3  0  0  0  0  0  0  0  0  0
     4          1         4  0  0  0  0  0  0  0  0  0
     5          1         5  0  0  0  0  0  0  0  0  0
     6          1         6  0  0  0  0  0  0  0  0  0
     7          1         7  0  0  0  0  0  0  0  0  0
     8          1         8  0  0  0  0  0  0  0  0  0
C-----------------------------------------------------------------------
```

3. Program Flow and Logic. (NAME2.DEF, PARM2.DEF and RETP2.FOR)

Input parameters are first read in by the program by calling the INPUT subroutine. The program then evoke the SIM subroutine which generates the random failure times and compute the key statistics required in the procedure. The next subroutine EVAL determines the measures of accuracy for run. REPORT is the subroutine which generates the output file for the run OUT2.DAT.

The variables in the program RETP2.FOR are described in the file NAME2.DEF as listed below.

```
C-----------------------------------------------------------------------
C  This file contains the declaration for input and output variables
C  used in the the RETP2 model.  (NAME2.DEF)  20 Jun 91
C-----------------------------------------------------------------------
C  Input Variables.
C  ---------------
C  ISEED = initial random seed selected.
C  SEED = current random seed.
C  RS = true overall series system reliability.
C  ALPHA = level of significance desired.
C  NREP = number of replications desired for the simulation.
C  TPN = test plan number (1).
C  TCN = test case number (1, 2, 3 or 4).
C  NCOMP = total number of components in the system.
C  NEXP = number of components with EXP failure times.
C  NWEI = number of components with WEI failure times.
C  NGEO = number of components with GEO failure times.
C  TOL = desired tolerance for MLE of WEI shape parameter.
C  Distribution:    EXPonential       WEIbull      GEOmetric
C  TY(I) = type:         1                2             3
C     PARM(1,I):     Scale(1/hr)     Scale(1/hr)     Prob
C     PARM(2,I):         -              Shape          -
C  UT(I) = utilization time (hrs) for component i (EXP and WEI).
C  UC(I) = utilization cycles for component i (GEO only).
C  NC(I) = number of test samples (sample size) for cmponent i.
C  NF(I) = desired number of failures in test for component i.
C  NCS = number of cut-sets for the system.
C  COMP(J,K) = kth parameter of cut-set j (first being the no. of
C              components belonging to the cut-set).
C
C  Assumed Variables.
C  -----------------
C  MAXCOMP = maximum number of components allowed in the system.
C  MAXREP = maximum number of replications permitted.
C  MAXCUT = maximum number of cut-sets.
C
C  Program and Output Variables.
C  ----------------------------
C  RS = true overall system reliability.
C  TT(I) = total accumulated failure time (hr) for component i
C          (EXP and WEI only).
C  TC(I) = total accumulated cycles to failure (incl. failure cycle)
C          for component i (GEO only).
C  EBETA(I) = estimate for shape parameter of component i (if Weibull).
C  REL1(J) = actual reliability for cut-set j.
C  REL2(J) = computed reliability for cut-set j for current replication.
C  ELM(I) = estimated component failure rate (1/hrs) for component i.
C  ELMAX(M) = max estimated component failure rate for rep m (1/hrs).
C  ER(I) = ratio of estimated failure rate to LMAX.
C  ET(I) = same as TT(I) except that these are for Weibull components.
```

```
C  NFC(M) = total number of failed test components.
C  LMU(M) = upper confidence bound for failure rate (1/hrs),
C  RSL(M) = lower confidence limit estimated for system reliability
C           ... for the mth replication.
C  ORSL(M) = ordered RSL(M) (ascending).
C  RSLOW = (1-ALPHA)x100 percentile of set of RSL(M).
C  LEVEL = achieved confidence level, ie. proportion of RSL(M) that
C          are lesser than RS (conservative estimate).
C
C-- END OF NAME2.DEF ------------------------------------------------
C
```

Together with the main program in RETP2.FOR are the other subroutines needed in the simulation. The declaration of variables is done in the file PARM2.DEF. Relevant descriptions are included as comment lines in the source code to help explain the program segments. A listing of PARM2.DEF and RETP2.FOR is given below.

```
C-------------------------------------------------------------------
C  This file contains the declaration for input and output variables
C  used in the the RETP2 model.   (PARM2.DEF)  20 Jun 91
C-------------------------------------------------------------------
      INTEGER MAXCOMP, MAXREP
      PARAMETER( MAXCOMP = 100 , MAXREP = 1000 , MAXCUT = 20 )
      REAL*8 ISEED, SEED
      INTEGER NREP, TCN, NCOMP, NEXP, NWEI, NGEO, NCS,
     *        NC(MAXCOMP), NF(MAXCOMP), TY(MAXCOMP), NFC(MAXREP),
     *        UC(MAXCOMP), TC(MAXCOMP), COMP(MAXCUT,MAXCOMP)
      REAL*8 RS, ALPHA, UT(MAXCOMP), TT(MAXCOMP),
     *       PARM(2,MAXCOMP), ELM(MAXCOMP), ER(MAXCOMP), ET(MAXCOMP),
     *       LMU(MAXREP), RSL(MAXREP), ORSL(MAXREP),
     *       ELMAX(MAXREP), RSLOW, LEVEL, TOL, EBETA(MAXCOMP),
     *       REL1(MAXCUT), REL2(MAXCUT)
C
      COMMON/BLOCK1/ISEED, SEED, NREP, TCN, NCOMP, NC, NF, NEXP, NWEI,
     *             NGEO, NCS, TY, NFC, UC, TC, COMP
      COMMON/BLOCK2/RS, ALPHA, UT, TT, PARM, ELM, ER, ET, LMU,
     *             RSL, ORSL, ELMAX, RSLOW, LEVEL, TOL, EBETA,
     *             REL1, REL2
C
C-- END OF PARM2.DEF ------------------------------------------------
C
C-------------------------------------------------------------------
C  This file contains the main program and the subroutines
C  for the Reliability Estimation Test Plan 2 (RETP2) model.
C  (RETP2.FOR) - runs on a IBM PC Compatible.
C
C  IBM Mainframe Version.
C
```

```
C  Test Plan 2 : Testing until accumulated time or cycles is achieved
C  -----------   for component i.
C
C  by Yee Kah-Chee  SMC 2802.
C  20 Jun 91.
C----------------------------------------------------------------------
C  1.  Main Program (RETP2).
C----------------------------------------------------------------------
      PROGRAM RETP2
C
C  Include the declaration files.
C
      INCLUDE 'NAME2 DEF'
      INCLUDE 'PARM2 DEF'
C
C  Read in input data.
C
      CALL INPUT
C
C  Activate simulation.
C
      CALL SIM
C
C  Process and evaluate output data.
C
      CALL EVAL
C
C  Generate simulation report.
C
      CALL REPORT
C
      STOP
      END
C----------------------------------------------------------------------
C  2.  Input Initialisation Subroutine (INPUT).
C----------------------------------------------------------------------
      SUBROUTINE INPUT
C
C  This subroutine reads in the inputs for the RETP2 model.
C
C  Include the declaration file.
C
      INCLUDE 'PARM2 DEF'
C
      INTEGER I, J, K, DUM2(11)
      REAL*8 DUM1(7)
C
C  Read data from 'IN2.DAT' designated as logic unit 1.
C
      OPEN(UNIT=1)
```

```
C
      READ(1,10)
   10 FORMAT(1X,/////)
      READ(1,*) ISEED
      READ(1,*) NCOMP
      READ(1,*) NEXP
      READ(1,*) NWEI
      READ(1,*) NGEO
      READ(1,*) TOL
      READ(1,*) ALPHA
      READ(1,*) NREP
      READ(1,*) TCN
      READ(1,20)
   20 FORMAT(1X,///)
      READ(1,*) NCS
C
      READ(1,30)
   30 FORMAT(1X,////////)
C
      DO 50 K = 1, NCOMP
        READ(1,*) DUM1
C
        I = NINT(DUM1(1))
        TY(I) = NINT(DUM1(2))
C
        IF (TY(I).EQ.1) THEN
          PARM(1,I) = DUM1(3)
          PARM(2,I) = DUM1(4)
          UT(I) = DUM1(5)
          TT(I) = DUM1(6)
          NC(I) = NINT(DUM1(7))
          EBETA(I) = 0
        ELSEIF (TY(I).EQ.2) THEN
          PARM(1,I) = DUM1(3)
          PARM(2,I) = DUM1(4)
          UT(I) = DUM1(5)
          TT(I) = DUM1(6)
          NC(I) = NINT(DUM1(7))
        ELSEIF (TY(I).EQ.3) THEN
          PARM(1,I) = DUM1(3)
          PARM(2,I) = DUM1(4)
          UC(I) = NINT(DUM1(5))
          UT(I) = DUM1(5)
          TT(I) = DUM1(6)
          NC(I) = NINT(DUM1(7))
        ENDIF
   50   CONTINUE
C
      READ(1,60)
   60 FORMAT(1X,///////////)
```

```
      DO 80 I = 1, NCS
        READ(1,*) DUM2
        J = DUM2(1)
        COMP(J,1) = DUM2(2)
        DO 70 K = 1, COMP(J,1)
          COMP(J,K+1) = DUM2(K+2)
   70   CONTINUE
   80 CONTINUE
      CLOSE(UNIT=1)
      RETURN
      END
C------------------------------------------------------------------------
C  3.  Subroutine (SIM)
C------------------------------------------------------------------------
      SUBROUTINE SIM
C
C  This subroutine simulates NREP possible outcomes of the test plan
C  desired in order to obtain the raw estimates of LMU(M) and RSL(M)
C  for each of the replication.
C
C  Include the declaration file
C  and declare local variables.
C
      INCLUDE 'PARM2 DEF'
      INTEGER I, J, K, M, ISUM, KEY, ICOUNT
      INTEGER NCYC(MAXCOMP), NSYS
      REAL*8 UNI
      REAL*8 SUM, PROD,
     *       FT(MAXCOMP,200), OFT(MAXCOMP,200),
     *       DFT(200), DOFT(200)
      LOGICAL FLAG
C
      SEED = ISEED
C
C  Compute overall true system reliability RS.
C
      RS = 1.0
      DO 30 J = 1, NCS
        PROD = 1.0
        DO 20 I = 1, COMP(J,1)
          K = COMP(J,I+1)
          PROD = PROD*( 1 - SURV(TY(K),PARM(1,K),PARM(2,K),UT(K)) )
   20   CONTINUE
        REL1(J) = 1.0 - PROD
        RS = RS * REL1(J)
   30 CONTINUE
C
C  Start of Simulation.
C  (Initialize replication counter M).
C
```

```
      M = 1
      DO WHILE (M.LE.NREP)
C
C  Test Plan : Sample and determine unknown NF(I)
C  ---------   (with known NC(I), Weibull case) until TT(I) is reached.
C
C  Generate NC(I) failure times, put them in ascending order
C  with the smallest failure time on the top of the list.
C
        DO 60 I = 1, NCOMP
C
C  Exponential components.  Test each component until it fails, check
C  to see if TT(I) is exceeded, if not, carry on testing.
C
          IF(TY(I).EQ.1) THEN
            SUM = 0.0
            L = 1
            DO WHILE (SUM.LE.TT(I))
              CALL LRNDPC(SEED,UNI,1)
              FT(I,L) = -LOG(UNI)/PARM(1,I)
              SUM = SUM + FT(I,L)
              L = L + 1
            ENDDO
            NF(I) = L - 2
C
C  Weibull components.  Generate NC(I) failure times, put them in
C  an ascending order with the smallest failure time on top and
C  determine NF(I).
C
          ELSEIF(TY(I).EQ.2) THEN
            DO 40 K = 1, NC(I)
              CALL LRNDPC(SEED,UNI,1)
                FT(I,K) = (1.0/PARM(1,I))*(-LOG(UNI))**(1.0/PARM(2,I))
                DFT(K) = FT(I,K)
  40        CONTINUE
            CALL BUBBLE(NC(I),DFT,DOFT)
            DO 42 K = 1, NC(I)
              OFT(I,K) = DOFT(K)
  42        CONTINUE
            SUM = 0.0
            NF(I) = 0
            DO 45 K = 1, NC(I)
              IF (OFT(I,K).LT.TT(I)) THEN
                NF(I) = NF(I) + 1
              ENDIF
  45        CONTINUE
C
C  GEOMETRIC COMPONENTS.  DETERMINE NF(I).
C
          ELSEIF(TY(I).EQ.3) THEN
```

```
          NF(I) = 0
          DO 50 K = 1, TT(I)
            CALL LRNDPC(SEED,UNI,1)
            IF (UNI.GT.PARM(1,I)) THEN
              NF(I) = NF(I) + 1
            ENDIF
   50     CONTINUE
C
        ENDIF
   60 CONTINUE
C
C  Determine total number of failed test components as well as
C  checking for zero component failure or just failures from a
C  SINGLE component (FLAG will be set to .TRUE. if so).
C
      ISUM = 0
      ICOUNT = 0
      DO 70 I = 1, NCOMP
        IF (NF(I).GT.0) THEN
          ICOUNT = ICOUNT + 1
        ENDIF
        ISUM = ISUM + NF(I)
   70 CONTINUE
C
      NFC(M) = ISUM
C
      IF (ICOUNT.LE.1) THEN
        FLAG = .TRUE.
      ELSE
        FLAG = .FALSE.
      ENDIF
C
C  Case A : More than ONE component type experienced failures
C  ------   in the test.
C
      IF (.NOT.FLAG) THEN
C
C  Estimate the failure rate of each component.
C
        DO 90 I = 1, NCOMP
C
C  Exponential components.
C
          IF (TY(I).EQ.1) THEN
            ET(I) = TT(I)
            ELM(I) = FLOAT(NF(I))/TT(I)
C
C  Weibull components.
C
          ELSEIF (TY(I).EQ.2) THEN
```

```
              DO 75 K = 1, NC(I)
                 DOFT(K) = OFT(I,K)
  75          CONTINUE
              CALL MLESHAPE(DOFT,NC(I),IMAX(1,NF(I)),TOL,1.D0,EBETA(I))
              EBETA(I) = BN(NC(I))*EBETA(I)
              SUM = 0.0
              DO 80 K = 1, NF(I)
                SUM = SUM + OFT(I,K)**EBETA(I)
   80         CONTINUE
              ET(I) = FLOAT(NC(I)-NF(I))*TT(I)**EBETA(I) + SUM
              ELM(I) = FLOAT(NF(I))/ET(I)
C
C  Geometric components.
C
            ELSEIF (TY(I).EQ.3) THEN
              ET(I) = TT(I)
              ELM(I) = FLOAT(NF(I))/TT(I)
C
            ENDIF
C
   90     CONTINUE
C
C  Determine the maximum failure rate estimate
C  and identify that component.
C
          ELMAX(M) = 0.0
          KEY = 0
          DO 100 I = 1, NCOMP
            IF (ELM(I).GT.ELMAX(M)) THEN
              ELMAX(M) = ELM(I)
              KEY = I
            ENDIF
  100     CONTINUE
C
C  Compute the ratios of the failure rate estimates to their maximum.
C
          DO 110 I = 1, NCOMP
            ER(I) = ELM(I) / ELMAX(M)
  110     CONTINUE
C
C  Determination of LMU(M)
C
          SUM = 0.0
          DO 120 I = 1, NCOMP
              SUM = SUM + (ER(I)*ET(I))
  120     CONTINUE
C
          LMU(M) = CHISQD(1-ALPHA,NINT(1.3*2*(1+NFC(M))))/(2*SUM)
C
C  Compute estimate of overall reliability RSL(M) for the system.
```

```
C
      RSL(M) = 1.0
      DO 140 J = 1, NCS
        PROD = 1.0
        DO 130 I = 1, COMP(J,1)
          K = COMP(J,I+1)
          IF (TY(K).EQ.1) THEN
            PROD = PROD*(1 - SURV(TY(K),LMU(M)*ER(K),EBETA(K),UT(K)))
          ELSEIF (TY(K).EQ.2) THEN
            PROD = PROD*(1 - SURV(TY(K),(LMU(M)*ER(K))**(1./EBETA(K)),
     *                       EBETA(K),UT(K)))
          ELSEIF (TY(K).EQ.3) THEN
            PROD = PROD*(1 - SURV(TY(K),1.D0-LMU(M)*ER(K),0.D0,UT(K)))
          ENDIF
 130    CONTINUE
        REL2(J) = 1.0 - PROD
        RSL(M) = RSL(M) * REL2(J)
 140  CONTINUE
C
    ENDIF
C
C  Case B : Where there are at most ONE component which experienced
C  ------   failures during the test.
C
    IF (FLAG) THEN
C
C  Determine number of complete systems (NSYS) implied by the test
C  based on cut-set information after first determining the number
C  of mission cycles tested for each component (to the nearest
C  integer) (NCYC(I)).
C
      ISUM = 0
      DO 150 I = 1, NCOMP
        NCYC(I) = INT(TT(I)/UT(I))
        ISUM = ISUM + NCYC(I)
        ELM(I) = 0
        ER(I) = 0
        EBETA(I) = 1.0
 150  CONTINUE
      ELMAX(M) = 0
      LMU(M) = 0
C
      NSYS = ISUM
      ISUM = 0
      DO 170 J = 1, NCS
        ISUM = 0
        DO 160 I = 1, COMP(J,1)
          ISUM = ISUM + NCYC(COMP(J,I+1))
 160    CONTINUE
        ISUM = ISUM / COMP(J,1)
```

```
          NSYS = IMIN(NSYS,ISUM)
 170    CONTINUE
C
C  For zero failures in the test.
C
        IF (ICOUNT.EQ.0) THEN
          RSL(M) = ALPHA**(1.0/FLOAT(NSYS))
C
C  For failures experienced by a particular component type.
C
        ELSE
          CALL GETP(NSYS,ALPHA,RSL(M))
        ENDIF
C
      ENDIF
C
C  Increment replication counter.
C
        M = M + 1
C
      ENDDO
      END
C-----------------------------------------------------------------------
C  4.  Subroutine (EVAL).
C-----------------------------------------------------------------------
      SUBROUTINE EVAL
C
C  This subroutine calls BUBBLE to sort the array RSL(NREP) in
C  ascending order to get an ordered array ORSL(NREP).  It also
C  determine the estimate for RSLOW at the specified significance
C  level ALPHA and the value of LEVEL in which ORSL(LEVEL) is closest
C  to the true reliability RS.
C
C  Include the declaration files
C  and declare the local variables.
C
      INCLUDE 'PARM2 DEF'
      INTEGER INDEX
      REAL*8 DIFF
C
C  Order the array RSL(NREP) in ascending order.
C
      DO 10 M = 1, NREP
        ORSL(M) = RSL(M)
   10 CONTINUE
C
C  Bubble Sort.  Sink the larger of the pair.
C
      CALL BUBBLE(NREP,RSL,ORSL)
C
```

```
C  Determine the (1-ALPHA)% lower confidence bound for the system
C  reliability.
C
      RSLOW = ORSL(NINT(NREP*(1-ALPHA)))
C
C  Finding the % confidence level for the true reliability RS.
C  (ie. the proportion of RSL(M) lesser than RS)
C
      DIFF = 1.0
      INDEX = 0
      DO 200 M = 1, NREP
        IF (ABS(ORSL(M)-RS).LT.DIFF) THEN
          DIFF = ABS(ORSL(M)-RS)
          INDEX = M
        ENDIF
  200 CONTINUE
C
      LEVEL = FLOAT(INDEX)/NREP
C
C  Record evaluated parameters in RAW2.DAT (unit 2).
C
      OPEN(UNIT=2)
      WRITE(2,300)
  300 FORMAT(1X, '     M      LMU(M)    ELMAX(M)      RSL(M)',
     *          '     ORSL(M)    NFC(M)')
      DO 500 M = 1, NREP
        WRITE(2,400) M,LMU(M),ELMAX(M),RSL(M),ORSL(M),NFC(M)
  400   FORMAT(1X,I6,2F12.7,2F12.7,I10)
  500 CONTINUE
      CLOSE(UNIT=2)
C
      RETURN
      END
C-----------------------------------------------------------------------
C  5.  Report Generation Subroutine (REPORT).
C-----------------------------------------------------------------------
       SUBROUTINE REPORT
C
C  This subroutine record the simulation results into the 'OUT2.DAT'
C  file as logic unit 3.
C
C  Include the declaration files
C  and declare local variables.
C
      INCLUDE 'PARM2 DEF'
      INTEGER I, J, K, DUM(10)
C
C  Write to output file 'OUT2.DAT' designated as logic unit 3.
C
      OPEN(UNIT=3)
```

```
C
      WRITE(3,10)
      WRITE(3,20) NREP
      WRITE(3,25)
      WRITE(3,26)
      WRITE(3,30)
      WRITE(3,40)
      WRITE(3,50) ISEED,NCOMP,ALPHA,TOL,NCS,TCN
C
      WRITE(3,60)
      DO 200 I = 1, NCOMP
        WRITE(3,70) I,TY(I),PARM(1,I),PARM(2,I),UT(I),TT(I),NC(I),NF(I)
  200 CONTINUE
      WRITE(3,80)
      WRITE(3,90)
      DO 300 I = 1, NCOMP
        WRITE(3,100) I,NF(I),ET(I),ELM(I),ER(I),EBETA(I)
  300 CONTINUE
      WRITE(3,110)
      WRITE(3,120)
      DO 500 J = 1, NCS
        DO 400 K = 1, 10
          DUM(K) = COMP(J,K)
  400   CONTINUE
        WRITE(3,130) J,DUM,REL1(J),REL2(J)
  500 CONTINUE
      WRITE(3,140)
      WRITE(3,150) RS,ELMAX(NREP),LMU(NREP),RSLOW,LEVEL
C
   10 FORMAT(1X,'OUT2.DAT : Output File of the RETP2 simulation')
   20 FORMAT(1X,'           after ',I5,' replications',/)
   25 FORMAT(1X,'COMMENTS : 8 COMPONENT IN SERIES                  ')
   26 FORMAT(1X,'           DF =  NINT (1.3 * 2 * (1 + NFC))       ',/)
   30 FORMAT(1X,'Input Parameters:',/)
   40 FORMAT(1X,'     ISEED   NCOMP   ALPHA     TOL   NCS   TCN',/)
   50 FORMAT(1X,F10.1,I8,F8.4,F8.5,2I6,/)
   60 FORMAT(1X,'  I TY(I) PARM1(I) PARM2(I)    UT(I)    TT(I)   NC(I)',
     *          '   NF(I)',/)
   70 FORMAT(1X,I3,I6,2F9.5,2F9.2,2I8)
   80 FORMAT(1X,/,'Output Parameters for the LAST Replication:',/)
   90 FORMAT(1X,'  I NF(I)           ET(I)         ELM(I)          ER(I)',
     *          '      EBETA(I)',/)
  100 FORMAT(1X,I3,I6,E16.7,2F14.7,F14.7)
  110 FORMAT(1X,/,'Cut-Set Data:',/)
  120 FORMAT(1X,'  J  NUM    Component List         ',
     *          '       REL1       REL2(M)',/)
  130 FORMAT(1X,I3,I5,9I3,2F12.9)
  140 FORMAT(1X,/,'          RS    ELMAX(M)      LMU(M)',
     *           '         RSLOW       LEVEL',/)
  150 FORMAT(1X,5F12.7,/)
```

```
C
      CLOSE(UNIT=3)
C
      RETURN
      END
C-----------------------------------------------------------------------
C  This portion of the file contains functions and subroutines
C  used in the RETP2 model.
C        - 20 Jun 91
C        - by Yee Kah-Chee SMC 2802
C-----------------------------------------------------------------------
C  A.  Random Number Generating Subroutine (LRNDPC).
C      (Courtesy of Mr. David Lim Hung-Heng)
C-----------------------------------------------------------------------
      SUBROUTINE LRNDPC (DSEED,U,N)
      INTEGER            N, I
      REAL*8             U(N)
      REAL*8             D31M1, DSEED, D31
C     D31M1=2**31 - 1
C     D31  =2**31
      DATA D31M1/2147483647.D0/
      DATA D31  /2147483648.D0/
      DO 5 I=1,N
C        DSEED = DMOD(950706376.D0*DSEED,D31M1)
         DSEED = DMOD(16807.D0*DSEED,D31M1)
    5 U(I) = DSEED / D31
      RETURN
      END
C------------------------------------------------------------------------
C  B.  Survivability Function.
C------------------------------------------------------------------------
      FUNCTION SURV(TYPE,PAR1,PAR2,UTIL)
C
C  This function returns the survival probability of the component of
C  different types (TYPE) with scale (PAR1) and shape (PAR2) parameters
C  given the specified utilization times or cycles (UTIL).
C
      INTEGER TYPE, N
      REAL*8 PAR1, PAR2, UTIL
C
      IF (TYPE.EQ.1) THEN
        SURV = EXP(-(PAR1*UTIL))
      ELSEIF (TYPE.EQ.2) THEN
        SURV = EXP(-((PAR1*UTIL)**PAR2))
      ELSE
        N = NINT(UTIL)
        SURV = PAR1**N
      ENDIF
C
      END
```

```
C-----------------------------------------------------------------------
C  C.  Bubble Sort Routine in ASCENDING Order.
C-----------------------------------------------------------------------
      SUBROUTINE BUBBLE(N,LIST,OLIST)
C
C  This subroutine performs a bubble sort in increasing order (ie. sink
C  the greater numeral) for the first N terms in an array LIST and
C  returns the result in OLIST.
C
      LOGICAL DONE
      INTEGER N, K, PAIR
      REAL*8 LIST(*), OLIST(*)
C
C  Sink the larger of the pair.
C
      DO 50 K = 1, N
        OLIST(K) = LIST(K)
   50 CONTINUE
      PAIR = N - 1
      DONE = .FALSE.
      DO WHILE (.NOT.DONE)
        DONE = .TRUE.
        DO 100 K = 1, PAIR
          IF (OLIST(K).GT.OLIST(K+1)) THEN
            TEMP = OLIST(K)
            OLIST(K) = OLIST(K+1)
            OLIST(K+1) = TEMP
            DONE = .FALSE.
          ENDIF
  100   CONTINUE
        PAIR = PAIR - 1
      ENDDO
      END
C-----------------------------------------------------------------------
C  D.  Unbiasing Factor for Biased MLE for Weibull Shape Parameter.
C-----------------------------------------------------------------------
      FUNCTION BN(I)
C
C  This functon returns the value of the unbiased factor for the biased
C  maximum likelihood estimate of the shape parameter of a Weibull
C  distribution with a sample size of N.
C
      INTEGER I
      IF (I.LE.5) THEN
        BN = (I*0.699)/(5.0)
      ELSEIF (I.EQ.6) THEN
        BN = 0.752
      ELSEIF (I.EQ.7) THEN
        BN = 0.786
      ELSEIF (I.EQ.8) THEN
```

```
        BN = 0.82
      ELSEIF (I.EQ.9) THEN
        BN = 0.8395
      ELSEIF (I.EQ.10) THEN
        BN = 0.859
      ELSEIF (I.EQ.11) THEN
        BN = 0.871
      ELSEIF (I.EQ.12) THEN
        BN = 0.883
      ELSEIF (I.EQ.13) THEN
        BN = 0.892
      ELSEIF (I.EQ.14) THEN
        BN = 0.901
      ELSEIF (I.EQ.15) THEN
        BN = 0.9075
      ELSEIF (I.EQ.16) THEN
        BN = 0.914
      ELSEIF (I.EQ.17) THEN
        BN = 0.9185
      ELSEIF (I.EQ.18) THEN
        BN = 0.923
      ELSEIF (I.EQ.19) THEN
        BN = 0.927
      ELSEIF (I.EQ.20) THEN
        BN = 0.931
      ELSEIF (I.LE.25) THEN
        BN = 0.931+(I-20)*0.014/5.0
      ELSEIF (I.LE.30) THEN
        BN = 0.945+(I-25)*0.01/5.0
      ELSEIF (I.LE.40) THEN
        BN = 0.955+(I-30)*0.011/10.0
      ELSEIF (I.LE.60) THEN
        BN = 0.966+(I-40)*0.012/20.0
      ELSEIF (I.LE.80) THEN
        BN = 0.978+(I-60)*0.006/20.0
      ELSEIF (I.LE.100) THEN
        BN = 0.984+(I-80)*0.003/20.0
      ELSEIF (I.LE.120) THEN
        BN = 0.987+(I-100)*0.003/20.0
      ELSE
        BN = 1.0
      ENDIF
      RETURN
      END
C-----------------------------------------------------------------------
C  E.  Biased MLE of Weibull Shape Parameter.
C-----------------------------------------------------------------------
      SUBROUTINE MLESHAPE(T,N,R,DEL,B,BNEW)
C
C  This subroutine returns a biased estimator (BNEW) for a Weibull
```

```
C  shape parameter using the Newton-Raphson's Method of Successive
C  Approximation.  The data parameters consist of an ascending ordered
C  list of failure times (T), sample size (N), number of failed samples
C  (R), tolerance for convergence (DEL) and an initial estimate of the
C  shape parameter (B).
C
      LOGICAL DONE
      INTEGER N, R, I
      REAL*8 GFUNCT, GPRIME, B, BOLD, BNEW, T(*), DEL,
     *       TERM1, TERM2, TERM3, SUM1, SUM2, SUM3, SUM4, STEP
C
      BNEW = B
      DONE = .FALSE.
C
      DO WHILE (.NOT.DONE)
C
        DONE = .TRUE.
        TERM1 = FLOAT(N-R)*(T(R)**BNEW)
        TERM2 = FLOAT(N-R)*(T(R)**BNEW)*LOG(T(R))
        TERM3 = FLOAT(N-R)*(T(R)**BNEW)*LOG(T(R))*LOG(T(R))
        SUM1 = 0.0
        SUM2 = 0.0
        SUM3 = 0.0
        SUM4 = 0.0
C
        DO 50 I = 1, R
          SUM1 = SUM1 + T(I)**BNEW
          SUM2 = SUM2 + (T(I)**BNEW)*LOG(T(I))
          SUM3 = SUM3 + (T(I)**BNEW)*LOG(T(I))*LOG(T(I))
          SUM4 = SUM4 + LOG(T(I))
   50   CONTINUE
C
        GFUNCT = (SUM2+TERM2)/(SUM1+TERM1) - (1.0/BNEW)
     *           - (1.0/FLOAT(R))*SUM4
C
        GPRIME = (1.0/(SUM1+TERM1)**2)*( (SUM1+TERM1)*(SUM3+TERM3)
     *                                  - (SUM2+TERM2)**2 )
     *          + (1.0/BNEW**2)
C
C     PRINT 60, GFUNCT,GPRIME,BNEW
C  60 FORMAT(1X, 'GFUNCT =',F8.3,' GPRIME =',F8.3,' BNEW =',F8.3)
C
C  Control magnitude of the marching step towards convergence
C  as no more than 0.1.
C
        IF ((GFUNCT.LT.0) .AND. (GPRIME.GT.0)) THEN
          STEP = VMAX(-.1D0,(GFUNCT/GPRIME))
        ELSEIF ((GFUNCT.GT.0) .AND. (GPRIME.LT.0)) THEN
          STEP = VMAX(-.1D0,(GFUNCT/GPRIME))
        ELSE
```

```
          STEP = VMIN(.1D0,(GFUNCT/GPRIME))
        ENDIF
C
        BOLD = BNEW
        BNEW = BNEW - STEP
C
C  Check for convergence of the MLE for the shape parameter B.
C
       IF (ABS(BOLD-BNEW).GT.DEL) THEN
          DONE = .FALSE.
        ENDIF
C
C  Avoid overflow error due to large MLE value caused by small
C  GPRIME (slope) as GFUNCT approaches to near zero.
C  STOP WHEN MAGNITUDE OF BNEW EXCEEDS 7.
C
       IF (BNEW.GT.7.0) THEN
         BNEW = BOLD
         DONE = .TRUE.
       ENDIF
C
      ENDDO
      RETURN
      END
C------------------------------------------------------------------------
C  E.  Chi-Square Quantile Function.
C------------------------------------------------------------------------
      FUNCTION CHISQD(P,N)
C
C  Modified version of Algorithm 451 from Comunications of the ACM
C  Aug 1977 Vol.16 No.8 .
C
C  This function evaluates the quantile at the probability level P
C  (left tail area) for the Chi-square distribution with
C  N degrees of freedom.
C
      REAL*8 P
      REAL X
      INTEGER IF
      DIMENSION C(21), A(19)
      DATA C/ 1.565326E-3,
     *        1.060438E-3,
     *       -6.950356E-3,
     *       -1.323293E-2,
     *        2.277679E-2,
     *       -8.986007E-3,
     *       -1.513904E-2,
     *        2.530010E-3,
     *       -1.450117E-3,
     *        5.169654E-3,
```

```
     *      -1.153761E-2,
     *       1.128186E-2,
     *       2.607083E-2,
     *      -0.2237368,
     *       9.780499E-5,
     *      -8.426812E-4,
     *       3.125580E-3,
     *      -8.553069E-3,
     *       1.348028E-4,
     *       0.4713941,
     *       1.0000886 /
      DATA A/ 1.264616E-2,
     *      -1.425296E-2,
     *       1.400483E-2,
     *      -5.886090E-3,
     *      -1.091214E-2,
     *      -2.304527E-2,
     *       3.135411E-3,
     *      -2.728484E-4,
     *      -9.699681E-3,
     *       1.316872E-2,
     *       2.618914E-2,
     *      -0.2222222,
     *       5.406674E-5,
     *       3.483789E-5,
     *      -7.274761E-4,
     *       3.292181E-3,
     *      -8.729713E-3,
     *       0.4714045,
     *       1. /
      IF (N-2) 10, 20, 30
   10 CALL XFROMP(.5*(1.-P),X,IF)
      CHISQD = X
      CHISQD = CHISQD*CHISQD
      RETURN
   20 CHISQD = -2.*LOG(1.-P)
      RETURN
   30 F = N
      F1 = 1./F
      CALL XFROMP(P,X,IF)
      T  = X
      F2 = SQRT(F1)*T
      IF (N.GE.(2+INT(4.*ABS(T)))) GO TO 40
      CHISQD = (((((((C(1)*F2+C(2))*F2+C(3))*F2+C(4))*F2
     *         +C(5))*F2+C(6))*F2+C(7))*F1+((((((C(8)+C(9)*F2)*F2
     *         +C(10))*F2+C(11))*F2+C(12))*F2+C(13))*F2+C(14)))*F1+
     *         (((((C(15)*F2+C(16))*F2+C(17))*F2+C(18))*F2
     *         +C(19))*F2+C(20))*F2+C(21)
      GO TO 50
   40 CHISQD=(((A(1)+A(2)*F2)*F1+(((A(3)+A(4)*F2)*F2
```

```
     *       +A(5))*F2+A(6)))*F1+(((((A(7)+A(8)*F2)*F2+A(9))*F2
     *       +A(10))*F2+A(11))*F2+A(12)))*F1+(((((A(13)*F2
     *       +A(14))*F2+A(15))*F2+A(16))*F2+A(17))*F2*F2
     *       +A(18))*F2+A(19)
   50 CHISQD = CHISQD*CHISQD*CHISQD*F
      RETURN
      END
C------------------------------------------------------------------------
C  F.  Standard Normal Variate Computation Subroutine.
C------------------------------------------------------------------------
      SUBROUTINE XFROMP(P,X,IFAULT)
C
C  Algorithm AS 24 J.R.STAT.SOC. C. (1969) Vol.18. No.3.
C
C  This subroutine computes the standard normal deviate X for
C  the specified left tail area P.
C
      REAL*8 P
      DIMENSION A(5)
      DIMENSION CONNOR (17), HSTNGS(6)
      DATA CONNOR/ 8.0327350124E-17,
     *             1.4483264644E-15,
     *             2.4668270103E-14,
     *             3.9554295164E-13,
     *             5.9477940136E-12,
     *             8.3507027951E-11,
     *             1.0892221037E-9,
     *             1.3122532964E-8,
     *             1.4503852223E-7,
     *             1.4589169001E-6,
     *             1.3227513228E-5,
     *             1.0683760684E-4,
     *             7.5757575758E-4,
     *             4.6296296296E-3,
     *             2.3809523810E-2,
     *             0.1,
     *             0.33333333333 /
C
      DATA RTHFPI / 1.2533141373 /
C
      DATA RRT2PI / 0.3989422804 /
C
      DATA TERMIN / 1.0E-11 /
C
      DATA HSTNGS / 2.515517,
     *               0.802853,
     *               0.010328,
     *               1.432788,
     *               0.189269,
     *               0.001308 /
```

```
C
      IFAULT = 1
      IF ((P.LE.0.0).OR.(P.GE.1.0)) GO TO 100
      IFAULT = 0
C
C Get first approximation XO to deviate by Hastings' formula
C
      B = P
      IF(B.GT.0.5) B = 1.0 - B
C
      F = - LOG(B)
      E = SQRT(F+F)
      XO = -E + ((HSTNGS(3)*E+HSTNGS(2))*E+HSTNGS(1))/
     *     (((HSTNGS(6)*E+HSTNGS(5))*E+HSTNGS(4))*E+1.0)
      IF (XO.LT.0.0) GO TO 1
      XO = 0.0
      PO = 0.5
      X1 = -RTHFPI
      GO TO 7
C
C Find the area PO corresponding to XO
C
    1 Y = XO**2
      IF (XO.LE.-1.9) GO TO 3
      Y = -0.5*Y
C
C (1) series approximation
C
      PO = CONNOR(1)
      DO 2 L=2,17
    2 PO = PO*Y + CONNOR(L)
      PO = (PO*Y+1.0)*XO
      X1 = -(PO+RTHFPI)*EXP(-Y)
      PO = PO*RRT2PI + 0.5
      GO TO 7
C
C (2) continued fraction approximation
C
    3 Z = 1.0/Y
      A(2) = 1.0
      A(3) = 1.0
      A(4) = Z + 1.0
      A(5) = 1.0
      W = 2.0
C
    4 DO 6 L=1,3,2
      DO 5 J=1,2
      K = L + J
      KA = 7 - K
C
```

```
    5 A(K) = A(KA) + A(K)*W*Z
C
    6 W = W + 1.0
      APPRXU = A(2)/A(3)
      APPRXL = A(5)/A(4)
      C = APPRXU - APPRXL
      IF (C.GE.TERMIN) GO TO 4
      X1 = APPRXL/XO
      PO = -X1*RRT2PI*EXP(-0.5*Y)
C
C Get accurate value of deviate by Taylor Series
C (X1, X2, X3 are derivatives for the Taylor Series
C
    7 D = F + LOG(PO)
      X2 = XO*X1*X1 -X1
      X3 = X1**3 + 2.0*XO*X1*X2 -X2
      X = ((X3*D/3.0+X2)*D/2.0+X1)*D + XO
      IF (P.LE.0.5) GO TO 100
      X = -X
  100 RETURN
      END
C-----------------------------------------------------------------------
C  G.  MAXIMUM FUNCTIONS.
C-----------------------------------------------------------------------
      FUNCTION VMAX(X,Y)
      REAL*8 X, Y
      IF (X.GT.Y) THEN
        VMAX = X
      ELSE
        VMAX = Y
      ENDIF
      RETURN
      END
      FUNCTION IMAX(X,Y)
      INTEGER X, Y
      IF (X.GT.Y) THEN
        IMAX = X
      ELSE
        IMAX = Y
      ENDIF
      RETURN
      END
C-----------------------------------------------------------------------
C  H.  Minimum Functions.
C-----------------------------------------------------------------------
      FUNCTION VMIN(X,Y)
      REAL*8 X, Y
      IF (X.LT.Y) THEN
        VMIN = X
      ELSE
```

```
        VMIN = Y
      ENDIF
      RETURN
      END
C
      FUNCTION IMIN(I,J)
      INTEGER I, J
      IF (I.LT.J) THEN
        IMIN = I
      ELSE
        IMIN = J
      ENDIF
      RETURN
      END
C------------------------------------------------------------------------
C  I.  ROUTINE TO EVALUATE VALUE OF P.
C------------------------------------------------------------------------
      SUBROUTINE GETP(N,ALPHA,NEWP)
      INTEGER N
      REAL*8 ALPHA, OLDP, NEWP, TOL
      LOGICAL DONE
      OLDP = ALPHA**(1.0/FLOAT(N))
      NEWP = 1.0
      TOL = 0.0001
      DONE = .FALSE.
      DO WHILE (.NOT.DONE)
        GFUNCT = N*OLDP**(N-1) - (N-1)*OLDP**N - ALPHA
        GPRIME = N*(N-1)*OLDP**(N-2) - N*(N-1)*OLDP**(N-1)
        NEWP = OLDP - (GFUNCT/GPRIME)
        IF ((ABS(NEWP-OLDP).LE.TOL) .OR. (ABS(GFUNCT).LE.TOL)) THEN
          DONE = .TRUE.
        ENDIF
        OLDP = NEWP
      END DO
      RETURN
      END
```

4. Program Output. (OUT2.DAT)

The result for the simulation run based on the input parameters specified in IN2.DAT are computed and written to the file OUT2.DAT. A sample of this file is as follows.

```
 OUT2.DAT : Output File of the RETP2 simulation
            after  1000 replications

 COMMENTS : 8 COMPONENT IN SERIES
            DF =  NINT (1.3 * 2 * (1 + NFC))
```

Input Parameters:

ISEED	NCOMP	ALPHA	TOL	NCS	TCN
16807.0	8	0.2000	0.01000	8	3

I	TY(I)	PARM1(I)	PARM2(I)	UT(I)	TT(I)	NC(I)	NF(I)
1	1	0.00500	1.00000	5.00	5400.00	20	36
2	1	0.00500	1.00000	5.00	5400.00	20	23
3	1	0.00500	1.00000	5.00	5400.00	20	28
4	1	0.00500	1.00000	5.00	5400.00	20	25
5	2	0.01000	2.00000	15.00	2700.00	20	20
6	2	0.01000	2.00000	15.00	2700.00	20	20
7	2	0.01000	2.00000	15.00	2700.00	20	20
8	2	0.01000	2.00000	15.00	2700.00	20	20

Output Parameters for the LAST Replication:

I	NF(I)	ET(I)	ELM(I)	ER(I)	EBETA(I)
1	36	0.5400000E+04	0.0066667	1.0000000	0.0000000
2	23	0.5400000E+04	0.0042593	0.6388889	0.0000000
3	28	0.5400000E+04	0.0051852	0.7777778	0.0000000
4	25	0.5400000E+04	0.0046296	0.6944444	0.0000000
5	20	0.2378985E+05	0.0008407	0.1261042	1.5441284
6	20	0.4825114E+06	0.0000414	0.0062175	2.1482593
7	20	0.2539978E+05	0.0007874	0.1181112	1.5843514
8	20	0.3708000E+07	0.0000054	0.0008091	2.5186896

Cut-Set Data:

J	NUM	Component List								REL1	REL2(M)
1	1	1	0	0	0	0	0	0	0	0.975309908	0.955163479
2	1	2	0	0	0	0	0	0	0	0.975309908	0.971117675
3	1	3	0	0	0	0	0	0	0	0.975309908	0.964950144
4	1	4	0	0	0	0	0	0	0	0.975309908	0.968645990
5	1	5	0	0	0	0	0	0	0	0.977751195	0.927053154
6	1	6	0	0	0	0	0	0	0	0.977751195	0.981007099
7	1	7	0	0	0	0	0	0	0	0.977751195	0.923940539
8	1	8	0	0	0	0	0	0	0	0.977751195	0.993218899

RS	ELMAX(M)	LMU(M)	RSLOW	LEVEL
0.8269590	0.0066667	0.0091745	0.7754698	0.9940000

APPENDIX D : Evaluation of Subroutines and Functions

RANDOM NUMBER GENERATOR
(LRNDPC)
Evaluation

One thousand *uniform* random real numbers between 0 and 1 are generated using the random number generating routine LRNDPC. From these uniformly distributed numbers, 1000 *exponential* (with scale parameter 1) numbers and 1000 *Weibull* (with scale parameter 1 and shape parameter 2) numbers were generated.

Uniform Random Variate

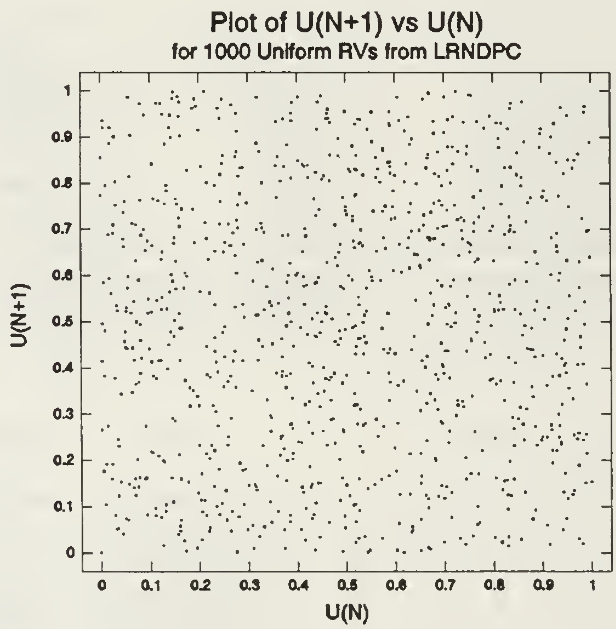

Figure 1 : Uniform RVs generated by LRNDPC

Figure 1 above shows a plot of 1000 *uniform* real numbers against their predecessors. The uniformity of the distribution of points over the state space confirms LRNDPC's adequacy in generating *uniform* random numbers.

Exponential and *Weibull* real numbers were generated using these 1000 Uniform(0,1) random numbers. The cumulative histograms of these resultant random variates were compared with their respective theoretical cumulative distribution functions (*cdf*s).

Exponential Random Variate

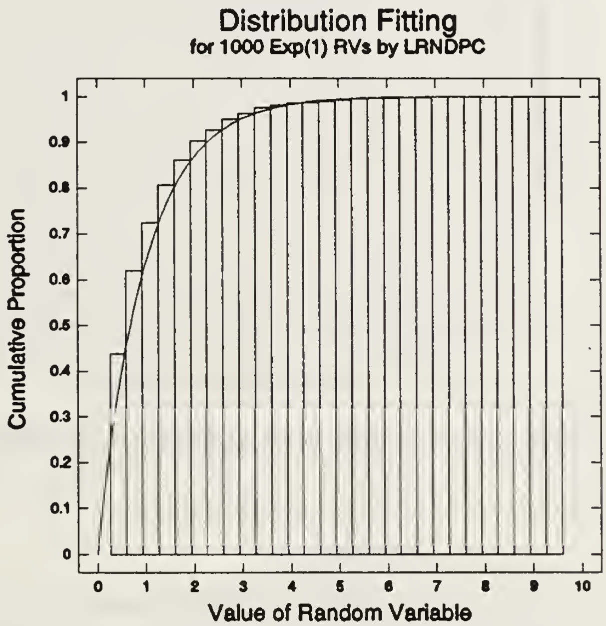

Figure 2 : Exponential RVs generated by LRNDPC

Figure 2 shows the close distribution fit between the theoretical cdf (line) and the cumulative distribution of *exponential* RV generated using LRNDPC.

$$\bar{F}(t) = \exp(-\lambda t)$$
$$\therefore \; t = -\frac{1}{\lambda}\ln\{\bar{F}(t)\}$$

Weibull Random Variate

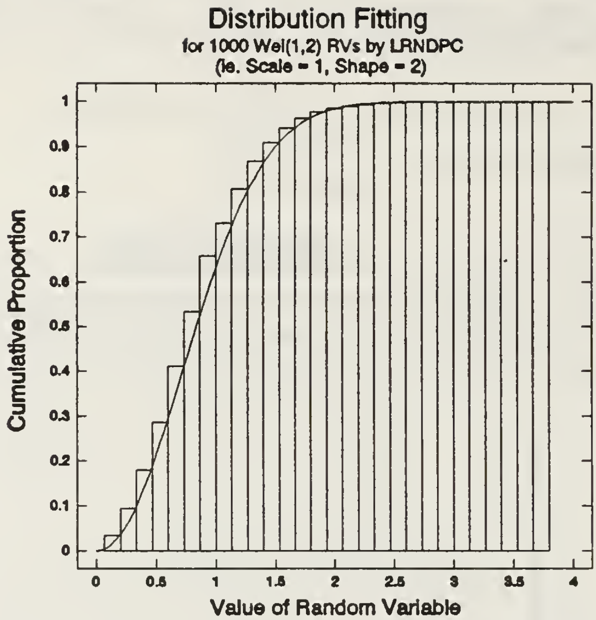

Figure 3 : Weibull RVs generated by LRNDPC

Figure 3 shows the close distribution fit between the theoretical cdf (line) and the cumulative distribution of *Weibull* RV generated using LRNDPC.

$$\bar{F}(t) = \exp\{-(\lambda t)^{\beta}\}$$

$$\therefore \quad t = -\frac{1}{\lambda}\left[\ln\{\bar{F}(t)\}\right]^{\frac{1}{\beta}}$$

Plot of Unbiasing Factor B(N) vs N
for Weibull Shape Parameter Estimation

N = Test Sample Size
B(N) = Unbiasing Factor for MLE

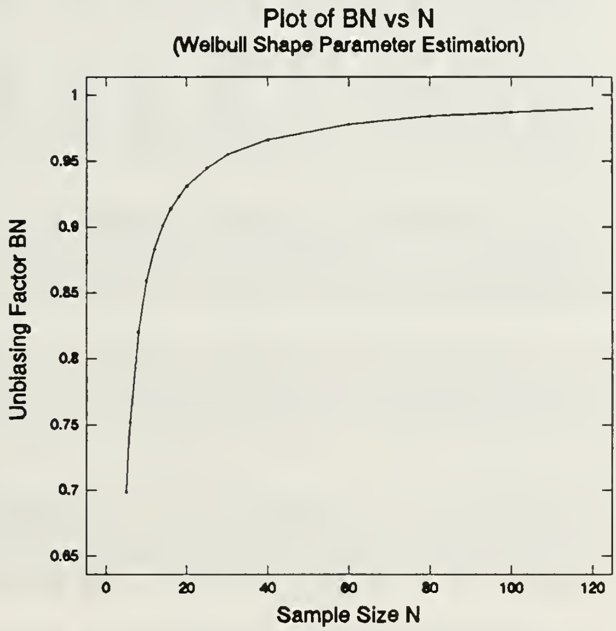

Figure 1 : B(N) vs N

The function BN(N) returns the linear-interpolated values of the unbiasing factor for the *raw* MLE $\hat{\beta}$ for both RETP1 and RETP2.

Evaluation of Subroutine CHISQD and XFROMP

The χ^2 statistics for 1 to 499 degrees of freedom for α values of 0.1 and 0.2 are generated using the routines CHISQD and XFROMP. These outputs matched those tabulated in the mathematical tables of any general textbook on statistics.

$$Prob[\ \chi^2_{df} \le table\ value\ ] = 1 - \alpha = 0.9$$

df	0	1	2	3	4	5	6	7	8	9
0		2.71	4.61	6.25	7.78	9.24	10.64	12.02	13.36	14.68
1	15.99	17.27	18.55	19.81	21.06	22.31	23.54	24.77	25.99	27.20
2	28.41	29.62	30.81	32.01	33.20	34.38	35.56	36.74	37.92	39.09
3	40.26	41.42	42.58	43.75	44.90	46.06	47.21	48.36	49.51	50.66
4	51.81	52.95	54.09	55.23	56.37	57.51	58.64	59.77	60.91	62.04
5	63.17	64.30	65.42	66.55	67.67	68.80	69.92	71.04	72.16	73.28
6	74.40	75.51	76.63	77.75	78.86	79.97	81.09	82.20	83.31	84.42
7	85.53	86.64	87.74	88.85	89.96	91.06	92.17	93.27	94.37	95.48
8	96.58	97.68	98.78	99.88	100.98	102.08	103.18	104.28	105.37	106.47
9	107.57	108.66	109.76	110.85	111.94	113.04	114.13	115.22	116.32	117.41
10	118.50	119.59	120.68	121.77	122.86	123.95	125.04	126.12	127.21	128.30
11	129.39	130.47	131.56	132.64	133.73	134.81	135.90	136.98	138.07	139.15
12	140.23	141.32	142.40	143.48	144.56	145.64	146.72	147.80	148.89	149.97
13	151.05	152.12	153.20	154.28	155.36	156.44	157.52	158.60	159.67	160.75
14	161.83	162.90	163.98	165.06	166.13	167.21	168.28	169.36	170.43	171.51
15	172.58	173.66	174.73	175.80	176.88	177.95	179.02	180.09	181.17	182.24
16	183.31	184.38	185.45	186.52	187.60	188.67	189.74	190.81	191.88	192.95
17	194.02	195.09	196.16	197.23	198.29	199.36	200.43	201.50	202.57	203.64
18	204.70	205.77	206.84	207.91	208.97	210.04	211.11	212.17	213.24	214.31
19	215.37	216.44	217.50	218.57	219.63	220.70	221.76	222.83	223.89	224.96
20	226.02	227.09	228.15	229.21	230.28	231.34	232.40	233.47	234.53	235.59
21	236.65	237.72	238.78	239.84	240.90	241.97	243.03	244.09	245.15	246.21
22	247.27	248.33	249.40	250.46	251.52	252.58	253.64	254.70	255.76	256.82
23	257.88	258.94	260.00	261.06	262.12	263.18	264.24	265.29	266.35	267.41
24	268.47	269.53	270.59	271.65	272.70	273.76	274.82	275.88	276.94	277.99
25	279.05	280.11	281.16	282.22	283.28	284.34	285.39	286.45	287.51	288.56
26	289.62	290.67	291.73	292.79	293.84	294.90	295.95	297.01	298.07	299.12
27	300.18	301.23	302.29	303.34	304.40	305.45	306.51	307.56	308.61	309.67
28	310.72	311.78	312.83	313.89	314.94	315.99	317.05	318.10	319.15	320.21
29	321.26	322.31	323.37	324.42	325.47	326.53	327.58	328.63	329.68	330.74
30	331.79	332.84	333.89	334.95	336.00	337.05	338.10	339.15	340.20	341.26
31	342.31	343.36	344.41	345.46	346.51	347.56	348.62	349.67	350.72	351.77
32	352.82	353.87	354.92	355.97	357.02	358.07	359.12	360.17	361.22	362.27
33	363.32	364.37	365.42	366.47	367.52	368.57	369.62	370.67	371.72	372.77
34	373.82	374.87	375.92	376.96	378.01	379.06	380.11	381.16	382.21	383.26
35	384.31	385.35	386.40	387.45	388.50	389.55	390.60	391.64	392.69	393.74
36	394.79	395.84	396.88	397.93	398.98	400.03	401.07	402.12	403.17	404.21
37	405.26	406.31	407.36	408.40	409.45	410.50	411.54	412.59	413.64	414.68
38	415.73	416.78	417.82	418.87	419.92	420.96	422.01	423.05	424.10	425.15
39	426.19	427.24	428.28	429.33	430.38	431.42	432.47	433.51	434.56	435.60
40	436.65	437.69	438.74	439.78	440.83	441.88	442.92	443.96	445.01	446.05
41	447.10	448.14	449.19	450.23	451.28	452.32	453.37	454.41	455.46	456.50
42	457.54	458.59	459.63	460.68	461.72	462.77	463.81	464.85	465.90	466.94
43	467.98	469.03	470.07	471.12	472.16	473.20	474.25	475.29	476.33	477.38
44	478.42	479.46	480.51	481.55	482.59	483.63	484.68	485.72	486.76	487.81
45	488.85	489.89	490.93	491.98	493.02	494.06	495.10	496.15	497.19	498.23
46	499.27	500.32	501.36	502.40	503.44	504.49	505.53	506.57	507.61	508.65
47	509.69	510.74	511.78	512.82	513.86	514.90	515.95	516.99	518.03	519.07
48	520.11	521.15	522.19	523.23	524.28	525.32	526.36	527.40	528.44	529.48
49	530.52	531.56	532.60	533.65	534.69	535.73	536.77	537.81	538.85	539.89

$$Prob[\ \chi^2_{df} \le \text{table value}\ ] = 1 - \alpha = 0.8$$

df	0	1	2	3	4	5	6	7	8	9
0		1.64	3.22	4.64	5.99	7.29	8.56	9.80	11.03	12.24
1	13.44	14.63	15.81	16.98	18.15	19.31	20.47	21.61	22.76	23.90
2	25.04	26.17	27.30	28.43	29.55	30.68	31.79	32.91	34.03	35.14
3	36.25	37.36	38.47	39.57	40.68	41.78	42.88	43.98	45.08	46.17
4	47.27	48.36	49.46	50.55	51.64	52.73	53.82	54.91	55.99	57.08
5	58.16	59.25	60.33	61.41	62.50	63.58	64.66	65.74	66.82	67.89
6	68.97	70.05	71.13	72.20	73.28	74.35	75.42	76.50	77.57	78.64
7	79.71	80.79	81.86	82.93	84.00	85.07	86.13	87.20	88.27	89.34
8	90.41	91.47	92.54	93.60	94.67	95.73	96.80	97.86	98.93	99.99
9	101.05	102.12	103.18	104.24	105.30	106.36	107.43	108.49	109.55	110.61
10	111.67	112.73	113.79	114.84	115.90	116.96	118.02	119.08	120.14	121.19
11	122.25	123.31	124.36	125.42	126.48	127.53	128.59	129.64	130.70	131.75
12	132.81	133.86	134.91	135.97	137.02	138.08	139.13	140.18	141.24	142.29
13	143.34	144.39	145.44	146.50	147.55	148.60	149.65	150.70	151.75	152.80
14	153.85	154.90	155.95	157.00	158.05	159.10	160.15	161.20	162.25	163.30
15	164.35	165.40	166.45	167.49	168.54	169.59	170.64	171.69	172.73	173.78
16	174.83	175.88	176.92	177.97	179.02	180.06	181.11	182.15	183.20	184.25
17	185.29	186.34	187.38	188.43	189.47	190.52	191.56	192.61	193.65	194.70
18	195.74	196.79	197.83	198.88	199.92	200.96	202.01	203.05	204.10	205.14
19	206.18	207.23	208.27	209.31	210.35	211.40	212.44	213.48	214.52	215.57
20	216.61	217.65	218.69	219.73	220.78	221.82	222.86	223.90	224.94	225.98
21	227.03	228.07	229.11	230.15	231.19	232.23	233.27	234.31	235.35	236.39
22	237.43	238.47	239.51	240.55	241.59	242.63	243.67	244.71	245.75	246.79
23	247.83	248.87	249.91	250.95	251.99	253.02	254.06	255.10	256.14	257.18
24	258.22	259.26	260.29	261.33	262.37	263.41	264.45	265.49	266.52	267.56
25	268.60	269.64	270.67	271.71	272.75	273.79	274.82	275.86	276.90	277.93
26	278.97	280.01	281.05	282.08	283.12	284.16	285.19	286.23	287.27	288.30
27	289.34	290.37	291.41	292.45	293.48	294.52	295.55	296.59	297.63	298.66
28	299.70	300.73	301.77	302.80	303.84	304.87	305.91	306.94	307.98	309.02
29	310.05	311.09	312.12	313.15	314.19	315.22	316.26	317.29	318.33	319.36
30	320.40	321.43	322.47	323.50	324.53	325.57	326.60	327.64	328.67	329.70
31	330.74	331.77	332.81	333.84	334.87	335.91	336.94	337.97	339.01	340.04
32	341.07	342.11	343.14	344.17	345.21	346.24	347.27	348.31	349.34	350.37
33	351.40	352.44	353.47	354.50	355.54	356.57	357.60	358.63	359.67	360.70
34	361.73	362.76	363.79	364.83	365.86	366.89	367.92	368.95	369.99	371.02
35	372.05	373.08	374.11	375.15	376.18	377.21	378.24	379.27	380.30	381.34
36	382.37	383.40	384.43	385.46	386.49	387.52	388.55	389.59	390.62	391.65
37	392.68	393.71	394.74	395.77	396.80	397.83	398.86	399.89	400.93	401.96
38	402.99	404.02	405.05	406.08	407.11	408.14	409.17	410.20	411.23	412.26
39	413.29	414.32	415.35	416.38	417.41	418.44	419.47	420.50	421.53	422.56
40	423.59	424.62	425.65	426.68	427.71	428.74	429.77	430.80	431.83	432.86
41	433.89	434.91	435.94	436.97	438.00	439.03	440.06	441.09	442.12	443.15
42	444.18	445.21	446.24	447.26	448.29	449.32	450.35	451.38	452.41	453.44
43	454.47	455.50	456.52	457.55	458.58	459.61	460.64	461.67	462.70	463.72
44	464.75	465.78	466.81	467.84	468.87	469.89	470.92	471.95	472.98	474.01
45	475.03	476.06	477.09	478.12	479.15	480.17	481.20	482.23	483.26	484.29
46	485.31	486.34	487.37	488.40	489.42	490.45	491.48	492.51	493.54	494.56
47	495.59	496.62	497.64	498.67	499.70	500.73	501.75	502.78	503.81	504.84
48	505.86	506.89	507.92	508.94	509.97	511.00	512.03	513.05	514.08	515.11
49	516.13	517.16	518.19	519.21	520.24	521.27	522.29	523.32	524.35	525.37

APPENDIX E : Tabulated Run Results for RETP1

Table 1A : 8 Exp in Series, RS = 0.931 (Hi)
min λ = 0.0002 f/hr, max λ = 0.0016 f/hr, UT = 10 hrs

S/N	Test Plan	Deg of Freedom	α	Measures of Accuracy	
				RSLOW	LEVEL
1	Test 5 until 5 failed. NFC=40	2*NFC (80)	0.1	0.919	0.982
			0.2	0.919	0.960
		2*(NFC+ NCOMP) (96)	0.1	0.906	1.000
			0.2	0.905	0.999
		2*NFC- NCOMP (72)	0.1	0.927	0.949
			0.2	0.927	0.880
		2*(NFC- NCOMP) (64)	0.1	0.934	0.821
			0.2	0.930	0.702
2	Test 15 until 15 failed. NFC=120	2*NFC (240)	0.1	0.928	0.955
			0.2	0.927	0.908
		2*(NFC+ NCOMP) (256)	0.1	0.923	0.999
			0.2	0.923	0.975
		2*NFC- NCOMP (232)	0.1	0.930	0.916
			0.2	0.934	0.833
		2*(NFC- NCOMP) (224)	0.1	0.932	0.844
			0.2	0.932	0.747
3	Test 15 until 11 failed. NFC=88	2*NFC (176)	0.1	0.927	0.955
			0.2	0.926	0.916
		2*(NFC+ NCOMP) (192)	0.1	0.921	0.916
			0.2	0.920	0.988
		2*NFC- NCOMP (168)	0.1	0.930	0.916
			0.2	0.927	0.830
		2*(NFC- NCOMP) (160)	0.1	0.933	0.843
			0.2	0.932	0.735

Table 1A : 8 Exp in Series, RS = 0.931 (Hi) (Cont...)
min λ = 0.0002 f/hr, max λ = 0.0016 f/hr, UT = 10 hrs

S/N	Test Plan	Deg of Freedom	α	Measures of Accuracy	
				RSLOW	LEVEL
4	Test 15 until 7 failed. NFC=56	2*NFC (112)	0.1	0.923	0.970
			0.2	0.923	0.931
		2*(NFC+ NCOMP) (128)	0.1	0.915	0.998
			0.2	0.913	0.994
		2*NFC- NCOMP (104)	0.1	0.929	0.919
			0.2	0.929	0.853
		2*(NFC- NCOMP) (96)	0.1	0.934	0.835
			0.2	0.933	0.720
5	Test 15 until 3 failed. NFC=24	2*NFC (48)	0.1	0.915	0.986
			0.2	0.912	0.975
		2*(NFC+ NCOMP) (64)	0.1	0.891	1.000
			0.2	0.888	1.000
		2*NFC- NCOMP (40)	0.1	0.927	0.944
			0.2	0.926	0.860
		2*(NFC- NCOMP) (32)	0.1	0.939	0.753
			0.2	0.939	0.634

Table 1B : 8 Exp in Series, RS = 0.803 (Lo)
min λ = 0.001 f/hr, max λ = 0.0045 f/hr, UT = 10 hrs

S/N	Test Plan	Deg of Freedom	α	Measures of Accuracy	
				RSLOW	LEVEL
1	Test 5 until 5 failed. NFC=40	2*NFC (80)	0.1	0.773	0.986
			0.2	0.793	0.963
		2*(NFC+ NCOMP) (96)	0.1	0.738	1.000
			0.2	0.736	0.999
		2*NFC- NCOMP (72)	0.1	0.798	0.858
			0.2	0.792	0.892
		2*(NFC- NCOMP) (64)	0.1	0.811	0.831
			0.2	0.812	0.713
2	Test 15 until 15 failed. NFC=120	2*NFC (240)	0.1	0.794	0.962
			0.2	0.793	0.916
		2*(NFC+ NCOMP) (256)	0.1	0.783	0.993
			0.2	0.798	0.981
		2*NFC- NCOMP (232)	0.1	0.800	0.924
			0.2	0.799	0.840
		2*(NFC- NCOMP) (224)	0.1	0.806	0.858
			0.2	0.805	0.755
3	Test 15 until 11 failed. NFC=88	2*NFC (176)	0.1	0.792	0.986
			0.2	0.790	0.921
		2*(NFC+ NCOMP) (192)	0.2	0.776	0.996
			0.2	0.774	0.981
		2*NFC- NCOMP (168)	0.1	0.800	0.922
			0.2	0.798	0.855
		2*(NFC- NCOMP) (160)	0.1	0.806	0.855
			0.2	0.806	0.746

Table 1B : 8 Exp in Series, RS = 0.803 (Lo) (Cont...)
min λ = 0.001 f/hr, max λ = 0.0045 f/hr, UT = 10 hrs

S/N	Test Plan	Deg of Freedom	α	Measures of Accuracy	
				RSLOW	LEVEL
4	Test 15 until 7 failed. NFC=56	2*NFC (112)	0.1	0.785	0.974
			0.2	0.782	0.938
		2*(NFC+NCOMP) (128)	0.1	0.760	0.998
			0.2	0.756	0.996
		2*NFC-NCOMP (104)	0.1	0.798	0.925
			0.2	0.795	0.860
		2*(NFC-NCOMP) (96)	0.1	0.811	0.841
			0.2	0.808	0.727
5	Test 15 until 3 failed. NFC=24	2*NFC (48)	0.1	0.759	0.977
			0.2	0.755	0.977
		2*(NFC+NCOMP) (64)	0.1	0.700	1.000
			0.2	0.793	1.000
		2*NFC-NCOMP (40)	0.1	0.791	0.949
			0.2	0.789	0.872
		2*(NFC-NCOMP) (32)	0.1	0.825	0.763
			0.2	0.825	0.642

Table 2A : 8 Wei in Series, RS = 0.980 (Hi)
min λ = 0.001 f/hr, max λ = 0.008 f/hr, UT = 10 hrs

S/N	Test Plan	Deg of Freedom	α	Measures of Accuracy	
				RSLOW	LEVEL
1	Test 5 until 5 failed. NFC=40	2*NFC (80)	0.1	0.977	0.992
			0.2	0.930	0.989
		2*(NFC+ NCOMP) (96)	0.1	0.937	0.994
			0.2	0.943	0.993
		2*NFC- NCOMP (72)	0.1	0.951	0.989
			0.2	0.937	0.980
		2*(NFC- NCOMP) (64)	0.1	0.956	0.985
			0.2	0.943	0.981
2	Test 15 until 15 failed. NFC=120	2*NFC (240)	0.1	0.978	0.918
			0.2	0.974	0.913
		2*(NFC+ NCOMP) (256)	0.1	0.977	0.931
			0.2	0.972	0.924
		2*NFC- NCOMP (232)	0.1	0.979	0.914
			0.2	0.975	0.901
		2*(NFC- NCOMP) (224)	0.1	0.983	0.904
			0.2	0.975	0.861
3	Test 15 until 11 failed. NFC=88	2*NFC (176)	0.1	0.982	0.876
			0.2	0.977	0.860
		2*(NFC+ NCOMP) (192)	0.1	0.980	0.914
			0.2	0.979	0.882
		2*NFC- NCOMP (168)	0.1	0.983	0.861
			0.2	0.978	0.839
		2*(NFC- NCOMP) (160)	0.1	0.978	0.840
			0.2	0.979	0.819

Table 2A : 8 Wei in Series, RS = 0.980 (Hi) (Cont...)
min λ = 0.001 f/hr, max λ = 0.008 f/hr, UT = 10 hrs

S/N	Test Plan	Deg of Freedom	α	Measures of Accuracy	
				RSLOW	LEVEL
4	Test 15 until 7 failed. NFC=56	2*NFC (112)	0.1	0.987	0.800
			0.2	0.981	0.779
		2*(NFC+ NCOMP) (128)	0.1	0.985	0.839
			0.2	0.978	0.824
		2*NFC- NCOMP (104)	0.1	0.988	0.776
			0.2	0.982	0.753
		2*(NFC- NCOMP) (96)	0.1	0.989	0.746
			0.2	0.989	0.732
5	Test 15 until 3 failed. NFC=24	2*NFC (48)	0.1	0.991	0.621
			0.2	0.991	0.584
		2*(NFC+ NCOMP) (64)	0.1	0.993	0.705
			0.2	0.988	0.685
		2*NFC- NCOMP (40)	0.1	0.995	0.548
			0.2	0.992	0.514
		2*(NFC- NCOMP) (32)	0.1	0.996	0.468
			0.2	0.993	0.417

Table 2B : 8 Wei in Series, RS = 0.832 (Lo)
min λ = 0.003 f/hr, max λ = 0.024 f/hr, UT = 10 hrs

S/N	Test Plan	Deg of Freedom	α	Measures of Accuracy	
				RSLOW	LEVEL
1	Test 5 until 5 failed. NFC=40	2*NFC (80)	0.1	0.736	0.985
			0.2	0.688	0.983
		2*(NFC+ NCOMP) (96)	0.1	0.696	0.992
			0.2	0.641	0.991
		2*NFC- NCOMP (72)	0.1	0.757	0.980
			0.2	0.713	0.973
		2*(NFC- NCOMP) (64)	0.1	0.778	0.968
			0.2	0.739	0.954
2	Test 15 until 15 failed. NFC=120	2*NFC (240)	0.1	0.834	0.876
			0.2	0.812	0.876
		2*(NFC+ NCOMP) (256)	0.1	0.825	0.920
			0.2	0.801	0.963
		2*NFC- NCOMP (232)	0.1	0.837	0.882
			0.2	0.817	0.858
		2*(NFC- NCOMP) (224)	0.1	0.844	0.866
			0.2	0.823	0.838
3	Test 15 until 11 failed. NFC=88	2*NFC (176)	0.1	0.854	0.838
			0.2	0.831	0.838
		2*(NFC+ NCOMP) (192)	0.1	0.842	0.882
			0.2	0.817	0.861
		2*NFC- NCOMP (168)	0.1	0.860	0.809
			0.2	0.837	0.776
		2*(NFC- NCOMP) (160)	0.1	0.865	0.777
			0.2	0.844	0.734

Table 2B : 8 Wei in Series, RS = 0.832 (Lo) (Cont...)
min λ = 0.003 f/hr, max λ = 0.024 f/hr, UT = 10 hrs

S/N	Test Plan	Deg of Freedom	α	Measures of Accuracy	
				RSLOW	LEVEL
4	Test 15 until 7 failed. NFC=56	2*NFC (112)	0.1	0.878	0.738
			0.2	0.857	0.708
		2*(NFC+ NCOMP) (128)	0.1	0.864	0.804
			0.2	0.839	0.780
		2*NFC- NCOMP (104)	0.1	0.886	0.702
			0.2	0.864	0.655
		2*(NFC- NCOMP) (96)	0.1	0.894	0.642
			0.2	0.875	0.587
5	Test 15 until 3 failed. NFC=24	2*NFC (48)	0.1	0.910	0.611
			0.2	0.888	0.560
		2*(NFC+ NCOMP) (64)	0.1	0.885	0.752
			0.2	0.856	0.719
		2*NFC- NCOMP (40)	0.1	0.923	0.515
			0.2	0.905	0.462
		2*(NFC- NCOMP) (32)	0.1	0.936	0.390
			0.2	0.922	0.306

Table 3A : 4 Exp and 4 Wei (Mixed) in Series, RS = 0.980 (Hi)
min λ = 0.002 f/hr, max λ = 0.008 f/hr, UT = 10 hrs

S/N	Test Plan	Deg of Freedom	α	Measures of Accuracy	
				RSLOW	LEVEL
1	Test 5 until 5 failed. NFC=40	2*NFC (80)	0.1	0.979	0.942
			0.2	0.978	0.905
		2*(NFC+ NCOMP) (96)	0.1	0.975	0.987
			0.2	0.981	0.976
		2*NFC- NCOMP (72)	0.1	0.981	0.881
			0.2	0.980	0.698
		2*(NFC- NCOMP) (64)	0.1	0.980	0.771
			0.2	0.982	0.684
2	Test 15 until 15 failed. NFC=120	2*NFC (240)	0.1	0.981	0.863
			0.2	0.980	0.800
		2*(NFC+ NCOMP) (256)	0.1	0.981	0.771
			0.2	0.980	0.898
		2*NFC- NCOMP (232)	0.1	0.981	0.881
			0.2	0.980	0.805
		2*(NFC- NCOMP) (224)	0.1	0.982	0.725
			0.2	0.981	0.631
3	Test 15 until 11 failed. NFC=88	2*NFC (176)	0.1	0.981	0.864
			0.2	0.981	0.801
		2*(NFC+ NCOMP) (192)	0.1	0.979	0.951
			0.2	0.978	0.907
		2*NFC- NCOMP (168)	0.1	0.982	0.698
			0.2	0.980	0.698
		2*(NFC- NCOMP) (160)	0.1	0.982	0.702
			0.2	0.982	0.591

Table 3A : 4 Exp and 4 Wei (Mixed) in Series, RS = 0.980 (Hi) (Cont...)
min λ = 0.002 f/hr, max λ = 0.008 f/hr, UT = 10 hrs

S/N	Test Plan	Deg of Freedom	α	Measures of Accuracy	
				RSLOW	LEVEL
4	Test 15 until 7 failed. NFC=56	2*NFC (112)	0.1	0.981	0.865
			0.2	0.980	0.787
		2*(NFC+ NCOMP) (128)	0.1	0.978	0.952
			0.1	0.978	0.920
		2*NFC- NCOMP (104)	0.1	0.982	0.769
			0.2	0.975	0.676
		2*(NFC- NCOMP) (96)	0.1	0.983	0.644
			0.2	0.983	0.523
5	Test 15 until 3 failed. NFC=24	2*NFC (48)	0.1	0.981	0.843
			0.2	0.981	0.762
		2*(NFC+ NCOMP) (64)	0.1	0.976	0.970
			0.2	0.975	0.941
		2*NFC- NCOMP (40)	0.1	0.984	0.684
			0.2	0.984	0.580
		2*(NFC- NCOMP) (32)	0.1	0.987	0.459
			0.2	0.987	0.356

Table 3B : 4 Exp and 4 Wei (Mixed) in Series, RS = 0.809 (Lo)
min λ = 0.002 f/hr, max λ = 0.008 f/hr, UT = 10 hrs

S/N	Test Plan	Deg of Freedom	α	Measures of Accuracy	
				RSLOW	LEVEL
1	Test 5 until 5 failed. NFC=40	2*NFC (80)	0.1	0.788	0.961
			0.2	0.777	0.930
		2*(NFC+ NCOMP) (96)	0.1	0.754	0.996
			0.2	0.794	0.987
		2*NFC- NCOMP (72)	0.1	0.805	0.910
			0.2	0.796	0.874
		2*(NFC- NCOMP) (64)	0.1	0.823	0.838
			0.2	0.815	0.741
2	Test 15 until 15 failed. NFC=120	2*NFC (240)	0.1	0.808	0.909
			0.2	0.805	0.842
		2*(NFC+ NCOMP) (256)	0.1	0.797	0.962
			0.2	0.794	0.987
		2*NFC- NCOMP (232)	0.1	0.814	0.854
			0.2	0.819	0.778
		2*(NFC- NCOMP) (224)	0.1	0.819	0.787
			0.2	0.817	0.787
3	Test 15 until 11 failed. NFC=88	2*NFC (176)	0.1	0.811	0.885
			0.2	0.807	0.820
		2*(NFC+ NCOMP) (192)	0.1	0.797	0.962
			0.2	0.792	0.925
		2*NFC- NCOMP (168)	0.1	0.817	0.821
			0.2	0.814	0.987
		2*(NFC- NCOMP) (160)	0.1	0.826	0.741
			0.2	0.822	0.647

Table 3B : 4 Exp and 4 Wei (Mixed) in Series, RS = 0.809 (Lo) (Cont...)
min λ = 0.002 f/hr, max λ = 0.008 f/hr, UT = 10 hrs

S/N	Test Plan	Deg of Freedom	α	Measures of Accuracy	
				RSLOW	LEVEL
4	Test 15 until 7 failed. NFC=56	2*NFC (112)	0.1	0.814	0.872
			0.2	0.810	0.792
		2*(NFC+ NCOMP) (128)	0.1	0.792	0.963
			0.2	0.787	0.931
		2*NFC-NCOMP (104)	0.1	0.825	0.775
			0.2	0.822	0.685
		2*(NFC-NCOMP) (96)	0.1	0.836	0.663
			0.2	0.834	0.550
5	Test 15 until 3 failed. NFC=24	2*NFC (48)	0.1	0.825	0.836
			0.2	0.815	0.755
		2*(NFC+ NCOMP) (64)	0.1	0.780	0.970
			0.2	0.766	0.940
		2*NFC-NCOMP (40)	0.1	0.849	0.970
			0.2	0.842	0.553
		2*(NFC-NCOMP) (32)	0.1	0.874	0.437
			0.2	0.869	0.338

APPENDIX F : Tabulated Run Results for RETP2

Table 4A : 8 Exp in Series, RS = 0.961 (Hi)
λ = 0.001 f/hr, UT = 5 hrs

S/N	Degrees of Freedom	K / E[NFC] (TT)	α	Measures of Accuracy	
				RSLOW	LEVEL
1	2*(1+NFC)	0.25 / 1.2 (225)	0.1	0.958	0.851
			0.1	0.935	0.851
		0.5 / 2.4 (450)	0.1	0.957	0.857
			0.2	0.954	0.857
		1.0 / 4.8 (900)	0.1	0.957	0.941
			0.2	0.957	0.850
		2.0 / 9.6 (1800)	0.1	0.958	0.916
			0.2	0.960	0.850
		3.0 / 14.4 (2700)	0.1	0.959	0.916
			0.2	0.959	0.809
		4.0 / 19.2 (3600)	0.1	0.959	0.937
			0.2	0.960	0.843
		5.0 / 24 (4500)	0.1	0.960	0.826
			0.2	0.961	0.914
		10.0 / 48 (9000)	0.1	0.960	0.924
			0.2	0.960	0.809
		20.0 / 96 (18000)	0.1	0.960	0.914
			0.2	0.961	0.820
		30.0 / 144 (27000)	0.1	0.961	0.906
			0.2	0.961	0.804

Table 4A : 8 Exp in Series, RS = 0.961 (Hi) (Cont...)
λ = 0.001 f/hr, UT = 5 hrs

S/N	Degrees of Freedom	K / E[NFC] (TT)	α	Measures of Accuracy	
				RSLOW	LEVEL
2	1.3* 2*(1+NFC)	0.25 / 1.2 (225)	0.1	0.950	0.851
			0.2	0.935	0.851
		0.5 / 2.4 (450)	0.1	0.950	0.857
			0.2	0.941	0.857
		1.0 / 4.8 (900)	0.1	0.946	0.851
			0.2	0.935	0.941
		2.0 / 9.6 (1800)	0.1	0.949	0.989
			0.2	0.950	0.948
		3.0 / 14.4 (2700)	0.1	0.949	0.998
			0.2	0.949	0.969
		4.0 / 19.2 (3600)	0.1	0.948	0.998
			0.2	0.950	0.987
		5.0 / 24 (4500)	0.1	0.949	0.998
			0.2	0.949	0.985
		10.0 / 48 (9000)	0.1	0.949	1.000
			0.2	0.949	0.997
		20.0 / 96 (18000)	0.1	0.949	1.000
			0.2	0.950	1.000
		30.0 / 144 (27000)	0.1	0.950	1.000
			0.2	0.949	1.000

Table 4B : 8 Exp in Series, RS = 0.819 (Lo)
λ = 0.005 f/hr, UT = 5 hrs

S/N	Degrees of Freedom	K / E[NFC] (TT)	α	Measures of Accuracy	
				RSLOW	LEVEL
1	2*(1+NFC)	0.25 / 1.2 (45)	0.1	0.774	0.851
			0.2	0.702	0.851
		0.5 / 2.4 (90)	0.1	0.801	0.857
			0.2	0.788	0.857
		1.0 / 4.8 (180)	0.1	0.801	0.919
			0.2	0.803	0.850
		2.0 / 9.6 (360)	0.1	0.807	0.916
			0.2	0.814	0.847
		3.0 / 14.4 (540)	0.1	0.812	0.916
			0.2	0.812	0.809
		4.0 / 19.2 (720)	0.1	0.809	0.923
			0.2	0.817	0.840
		5.0 / 24 (900)	0.1	0.813	0.925
			0.2	0.816	0.814
		10.0 / 48 (1800)	0.1	0.816	0.919
			0.2	0.817	0.809
		20.0 / 96 (3600)	0.1	0.817	0.814
			0.2	0.818	0.820
		30.0 / 144 (5400)	0.1	0.817	0.907
			0.2	0.818	0.804

Table 4B : 8 Exp in Series, RS = 0.819 (Lo) (Cont...)
λ = 0.005 f/hr, UT = 5 hrs

S/N	Degrees of Freedom	K / E[NFC] (TT)	α	Measures of Accuracy	
				RSLOW	LEVEL
2	1.3* 2*(1+NFC)	0.25 / 1.2 (45)	0.1	0.774	0.851
			0.2	0.702	0.851
		0.5 / 2.4 (90)	0.1	0.801	0.857
			0.2	0.736	0.857
		1.0 / 4.8 (180)	0.1	0.759	0.981
			0.2	0.753	0.941
		2.0 / 9.6 (360)	0.1	0.762	0.989
			0.2	0.771	0.948
		3.0 / 14.4 (540)	0.1	0.770	0.991
			0.2	0.768	0.969
		4.0 / 19.2 (720)	0.1	0.766	0.997
			0.2	0.772	0.969
		5.0 / 24 (900)	0.1	0.769	0.998
			0.2	0.771	0.985
		10.0 / 48 (1800)	0.1	0.771	1.000
			0.2	0.771	0.997
		20.0 / 96 (3600)	0.1	0.771	1.000
			0.2	0.772	1.000
		30.0 / 144 (5400)	0.1	0.773	1.000
			0.2	0.772	1.000

Table 5A : 8 Wei in Series, RS = 0.956 (Hi) (*)
λ = 0.005 f/hr, UT = 15 hrs

S/N	Degrees of Freedom	K / E[NFC] (TT)	α	Measures of Accuracy	
				RSLOW	LEVEL
1	2*(1+NFC)	0.25 / 1.2 (45)	0.1	1.000	0.186
			0.2	1.000	0.158
		0.5 / 2.4 (90)	0.1	0.986	0.501
			0.2	0.979	0.458
		1.0 / 4.8 (180)	0.1	0.967	0.767
			0.2	0.960	0.732
		2.0 / 9.6 (360)	0.1	0.957	0.879
			0.2	0.952	0.854
		3.0 / 14.4 (540)	0.1	0.952	0.934
			0.2	0.946	0.922
		4.0 / 19.2 (720)	0.1	0.952	0.934
			0.2	0.946	0.928
		5.0 / 24 (900)	0.1	0.952	0.940
			0.2	0.946	0.928
		10.0 / 48 (1800)	0.1	0.952	0.940
			0.2	0.946	0.928
		20.0 / 96 (3600)	0.1	0.952	0.940
			0.2	0.946	0.928
		30.0 / 144 (5400)	0.1	0.952	0.940
			0.2	0.946	0.928

(*) 20 test items for each *Weibull* component.

Table 5A : 8 Wei in Series, RS = 0.956 (Hi) (*) (Cont...)
λ = 0.005 f/hr, UT = 15 hrs

S/N	Degrees of Freedom	K / E[NFC] (TT)	α	Measures of Accuracy	
				RSLOW	LEVEL
2	1.3* 2*(1+NFC)	0.25 / 1.2 (45)	0.1	1.000	0.258
			0.2	0.999	0.224
		0.5 / 2.4 (90)	0.1	0.983	0.635
			0.2	0.983	0.593
		190 / 4.8 (180)	0.1	0.958	0.884
			0.2	0.930	0.866
		2.0 / 9.6 (360)	0.1	0.946	0.963
			0.2	0.939	0.956
		3.0 / 14.4 (540)	0.1	0.939	0.984
			0.2	0.930	0.976
		4.0 / 19.2 (720)	0.1	0.938	0.987
			0.2	0.930	0.981
		5.0 / 24 (900)	0.1	0.938	0.987
			0.2	0.930	0.981
		10.0 / 48 (1800)	0.1	0.938	0.987
			0.2	0.946	0.981
		20.0 / 96 (3600)	0.1	0.938	0.987
			0.2	0.930	0.981
		30.0 / 144 (5400)	0.1	0.938	0.987
			0.2	0.930	0.981

Table 5B : 8 Wei in Series, RS = 0.835 (Lo) (*)
λ = 0.01 f/hr, UT = 15 hrs

S/N	Degrees of Freedom	K / E[NFC] (TT)	α	Measures of Accuracy	
				RSLOW	LEVEL
1	2*(1+NFC)	0.25 / 1.2 (22.5)	0.1	0.966	0.222
			0.2	0.963	0.161
		0.5 / 2.4 (45)	0.1	0.932	0.367
			0.2	0.913	0.319
		1.0 / 4.8 (90)	0.1	0.874	0.704
			0.2	0.858	0.658
		2.0 / 9.6 (180)	0.1	0.842	0.851
			0.2	0.832	0.819
		3.0 / 14.4 (270)	0.1	0.829	0.924
			0.2	0.814	0.908
		4.0 / 19.2 (360)	0.1	0.827	0.928
			0.2	0.813	0.908
		5.0 / 24 (450)	0.1	0.827	0.928
			0.2	0.813	0.908
		10.0 / 48 (900)	0.1	0.827	0.928
			0.2	0.813	0.908
		20.0 / 96 (1800)	0.1	0.827	0.928
			0.2	0.813	0.908
		30.0 / 144 (2700)	0.1	0.827	0.928
			0.2	0.813	0.908

Table 5B : 8 Wei in Series, RS = 0.835 (Lo) (*)
λ = 0.01 f/hr, UT = 15 hrs

S/N	Degrees of Freedom	K / E[NFC] (TT)	α	Measures of Accuracy	
				RSLOW	LEVEL
2	1.3* 2*(1+NFC)	0.25 / 1.2 (22.5)	0.1	0.957	0.326
			0.2	0.953	0.268
		0.5 / 2.4 (45)	0.1	0.914	0.572
			0.2	0.890	0.522
		1.0 / 4.8 (90)	0.1	0.842	0.881
			0.2	0.821	0.862
		2.0 / 9.6 (180)	0.1	0.802	0.968
			0.2	0.784	0.962
		3.0 / 14.4 (270)	0.1	0.786	0.986
			0.2	0.766	0.983
		4.0 / 19.2 (360)	0.1	0.784	0.989
			0.2	0.786	0.986
		5.0 / 24 (450)	0.1	0.784	0.989
			0.2	0.766	0.986
		10.0 / 48 (900)	0.1	0.784	0.989
			0.2	0.766	0.986
		20.0 / 96 (1800)	0.1	0.784	0.989
			0.2	0.766	0.986
		30.0 / 144 (2700)	0.1	0.784	0.989
			0.2	0.766	0.986

Table 6A : 4 Exp and 4 Wei (Mixed) in Series, RS = 0.958 (Hi) (*)
λ(exp) = 0.001 f/hr, UT(exp) = 5 hrs
λ(wei) = 0.005 f/hr, UT(wei) = 15 hrs

S/N	Degrees of Freedom	K / E[NFC] TT(exp) TT(wei)	α	Measures of Accuracy	
				RSLOW	LEVEL
1	2*(1+NFC)	0.25 / 1.2 (225) (45)	0.1	1.000	0.620
			0.2	0.995	0.451
		0.5 / 2.4 (450) (90)	0.1	0.960	0.623
			0.2	0.975	0.573
		1.0 / 4.8 (900) (180)	0.1	0.971	0.736
			0.2	0.965	0.684
		2.0 / 9.6 (1800) (360)	0.1	0.956	0.803
			0.2	0.960	0.765
		3.0 / 14.4 (2700) (540)	0.1	0.960	0.874
			0.2	0.956	0.841
		4.0 / 19.2 (3600) (720)	0.1	0.960	0.873
			0.2	0.959	0.839
		5.0 / 24 (4500) (900)	0.1	0.959	0.887
			0.2	0.956	0.861
		10.0 / 48 (9000) (1800)	0.1	0.959	0.891
			0.2	0.959	0.862
		20.0 / 96 (18000) (3600)	0.1	0.959	0.892
			0.2	0.955	0.867
		30.0 / 144 (27000) (5400)	0.1	0.956	0.892
			0.2	0.956	0.862

Table 6A : 4 Exp and 4 Wei (Mixed) in Series, RS = 0.958 (Hi) (*) (Cont...)
λ(exp) = 0.001 f/hr, UT(exp) = 5 hrs
λ(wei) = 0.005 f/hr, UT(wei) = 15 hrs

S/N	Degrees of Freedom	K / E[NFC] TT(exp) TT(wei)	α	Measures of Accuracy	
				RSLOW	LEVEL
2	1.3* 2*(1+NFC)	0.25 / 1.2 (225) (45)	0.1	1.000	0.723
			0.2	0.994	0.697
		0.5 / 2.4 (450) (90)	0.1	0.977	0.730
			0.2	0.969	0.681
		1.0 / 4.8 (900) (180)	0.1	0.964	0.853
			0.2	0.955	0.837
		2.0 / 9.6 (1800) (360)	0.1	0.954	0.934
			0.2	0.949	0.921
		3.0 / 14.4 (2700) (540)	0.1	0.949	0.961
			0.2	0.944	0.978
		4.0 / 19.2 (3600) (720)	0.1	0.948	0.975
			0.2	0.949	0.966
		5.0 / 24 (4500) (900)	0.1	0.948	0.984
			0.2	0.944	0.978
		10.0 / 48 (9000) (1800)	0.1	0.948	0.978
			0.2	0.943	0.965
		20.0 / 96 (18000) (3600)	0.1	0.949	0.978
			0.2	0.942	0.979
		30.0 / 144 (27000) (5400)	0.1	0.949	0.978
			0.2	0.944	0.971

Table 6B : 4 Exp and 4 Wei (Mixed) in Series, RS = 0.827 (Lo) (*)
λ(exp) = 0.005 f/hr, UT(exp) = 5 hrs
λ(wei) = 0.010 f/hr, UT(wei) = 15 hrs

S/N	Degrees of Freedom	K / E[NFC] TT(exp) TT(wei)	α	Measures of Accuracy	
				RSLOW	LEVEL
1	2*(1+NFC)	0.25 / 1.2 (45) (22.5)	0.1	0.971	0.691
			0.2	0.938	0.685
		0.5 / 2.4 (90) (45)	0.1	0.916	0.600
			0.2	0.893	0.532
		1.0 / 4.8 (180) (90)	0.1	0.876	0.684
			0.2	0.855	0.638
		2.0 / 9.6 (360) (180)	0.1	0.849	0.787
			0.2	0.836	0.750
		3.0 / 14.4 (540) (270)	0.1	0.838	0.855
			0.2	0.824	0.817
		4.0 / 19.2 (720) (360)	0.1	0.838	0.870
			0.2	0.829	0.819
		5.0 / 24 (900) (450)	0.1	0.829	0.890
			0.2	0.822	0.838
		10.0 / 48 (1800) (900)	0.1	0.827	0.900
			0.2	0.820	0.863
		20.0 / 96 (3600) (1800)	0.1	0.826	0.902
			0.2	0.819	0.871
		30.0 / 144 (5400) (2700)	0.1	0.831	0.684
			0.2	0.821	0.854

Table 6B : 4 Exp and 4 Wei (Mixed) in Series, RS = 0.827 (Lo) (*) (Cont...)
λ(exp) = 0.005 f/hr, UT(exp) = 5 hrs
λ(wei) = 0.010 f/hr, UT(wei) = 15 hrs

S/N	Degrees of Freedom	K / E[NFC] TT(exp) TT(wei)	α	Measures of Accuracy	
				RSLOW	LEVEL
2	1.3* 2*(1+NFC)	0.25 / 1.2 (45) (22.5)	0.1	0.964	0.718
			0.2	0.922	0.700
		0.5 / 2.4 (90) (45)	0.1	0.895	0.988
			0.2	0.866	0.688
		1.0 / 4.8 (180) (90)	0.1	0.846	0.864
			0.2	0.818	0.988
		2.0 / 9.6 (360) (180)	0.1	0.811	0.939
			0.2	0.798	0.921
		3.0 / 14.4 (540) (270)	0.1	0.798	0.983
			0.2	0.780	0.952
		4.0 / 19.2 (720) (360)	0.1	0.772	0.939
			0.2	0.780	0.972
		5.0 / 24 (900) (450)	0.1	0.787	0.988
			0.2	0.777	0.983
		10.0 / 48 (1800) (900)	0.1	0.783	0.688
			0.2	0.774	0.983
		20.0 / 96 (3600) (1800)	0.1	0.783	0.993
			0.2	0.772	0.988
		30.0 / 144 (5400) (2700)	0.1	0.788	0.996
			0.2	0.775	0.994

LIST OF REFERENCES

1. Mann, N., R. Schafer, N. Singpurwalla, (1974) *Methods of Statistical Analysis of Reliability and Life Data*, John Wiley and Sons.

2. Myhre, J., A. Rosenfeld and S. Saunders, (1978) *Naval Research Quarterly*, 25(2).

3. Bain, L. and M. Englehardt, *Introduction to Probability and Mathematical Statistics*, 1987, Duxbury Press.

4. Strategic Systems Project Office, (1978) NAVSEA OD29304B *Reliability and Availability Evaluation Program Manual.*

5. Bain, Lee, (1978) *Statiscal Analysis of Reliability and Life-Testing Models*, Marcel Decker.

INITIAL DISTRIBUTION LIST

		No. of Copies
1.	Defence Technical Information Center Cameron Station Alexandria, VA 22304-6145, USA	2
2.	Library, Code 52 Naval Postgraduate School Monterey, CA 93943-5002, USA	2
3.	Professor W. Max. Woods Naval Postgraduate School, Code OR/Wo Monterey, Ca 93943-5000, USA	7
4.	Professor R. R. Read Naval Postgraduate School, Code OR/Re Monterey, CA 93943-5000, USA	1
5.	Professor Wayne Hughes Naval Postgraduate School, Code OR/Hu Monterey, CA 93943-5000, USA	1
5.	YEE, Kah-Chee Defence Science Organisation Building 20, Science Park Drive Singapore 0511 REPUBLIC OF SINGAPORE	2

Thesis
Y3366 Yee
c.1 Approximate confidence
limit procedures for complex systems.